JAIEP

Automatic Term Extraction for Arabic Text: Approaches, Techniques, and Challenges

Mariam Muhammed [1, *], Shahira Azab [2], Nesrine Ali [1], Mervat Ghieth [2]

[1] Department of Information Systems and Technology, Faculty of Graduate Studies for Statistical Research (FGSSR), Cairo University, Giza, Egypt
[2] Department of Computer Science, Faculty of Graduate Studies for Statistical Research (FGSSR), Cairo University, Giza, Egypt
* Corresponding author: eng.maryamadel@cu.edu.eg

Abstract

Automatic Term Extraction (ATE) is an essential task in Natural Language Processing (NLP) that aims to identify domain-specific terms from large corpora. In the context of Arabic, ATE plays an essential role in applications such as ontology construction, dictionary development, information retrieval, and text mining. However, the rich morphological structure, and orthographic ambiguities of Arabic present unique challenges in the process of ATE. This paper provides a comprehensive survey of ATE for Arabic text, with a focus on approaches, techniques, and evaluation strategies. We review rule-based, statistical, machine learning, deep learning, and hybrid methods, reviewing their strengths, limitations, and applicability to Arabic's linguistic characteristics. We also review challenges that affect the ATE process such as morphological richness, multiword expression extraction, named entity recognition, and the scarcity of annotated corpora. Furthermore, we outline evaluation metrics that are essential for assessing performance in Arabic. This paper aims to support the development of more accurate, adaptable, and domain specific ATE systems for Arabic texts.

Keywords: Arabic Language, Automatic Term Extraction, Arabic Natural Language Processing, Multiword Expressions, Information Retrieval.
MSC: 03B65; 97F30

Doi: https://doi.org/10.21608/jaiep.2025.423298.1025
Received: September 13, 2025; Revised: October 13, 2025; Accepted: October 15, 2025

1. Introduction

Since it has a direct impact on how an organization conducts business, information security ought to be its top priority. There are both technical and non-technical aspects to information security. Technical security issues can be resolved by installing a firewall, antivirus software, backing up data, implementing access control measures, encrypting the system, and continuously monitoring it for threats. Measures of employee behavior are considered non-technical measures. Information security-related sociological, psychological, and organizational behavioral theories are included in these procedures to guarantee that employees follow information security policies [1].

Automatic Term Extraction (ATE) is an important task in Natural Language Processing (NLP) that focuses on identifying domain-specific terms, both single-word and multiword expressions, from large corpora [1]. These terms are essential for representing specific knowledge and play a necessary role in a wide range of applications, including ontology construction, dictionary development,

machine translation, information retrieval, and text mining The process of ATE typically involves three main stages [2], as it is shown in "Fig.1":

1. **Preprocessing and Term Candidate Extraction –** transforming input text into linguistic units such as tokens, lemmas, or n-grams to generate initial term candidates.
2. **Term Candidate Scoring –** assigning a numerical score (termhood) to each candidate based on statistical, linguistic, or hybrid criteria.
3. **Term Candidate Ranking and Selection –** ordering candidates according to their scores and selecting the top-ranked terms as the final domain-specific terminology.

Compared with other languages, Arabic presents unique challenges for ATE due to its rich morphology, orthographic ambiguity, and frequent use of multiword expressions [3]. For example, the root-based derivational system in Arabic produces various inflected forms from a single root, making it difficult to consistently identify related terms. Additionally, the absence of short vowels in written text introduces ambiguity in terms [4]. A detailed comparison between Arabic and English linguistic characteristics and their impact on ATE is provided in Appendix A.

Recent research has presented different approaches such as rule-based and statistical methods to machine learning, deep learning, and hybrid systems to address Arabic-specific linguistic complexities [2, 5, 6].

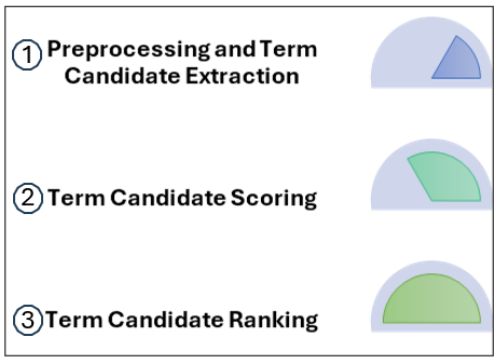

Figure 1: The stages of term extraction

This paper presents a comprehensive survey of ATE for Arabic text, with a focus on approaches, techniques, and challenges. We review existing methods, analyze their strengths and limitations, and discuss evaluation metrics that are critical for assessing performance in the Arabic context. This work aims to guide future developments in building accurate, adaptable, and domain-specific ATE systems for Arabic.

This survey primarily targets peer-reviewed publications, including journal articles and conference papers that address Arabic Term Extraction (ATE) and related Natural Language Processing (NLP) tasks such as Named Entity Recognition (NER) and Multi-Word Term (MWT) extraction [1, 3, 7–10]. Nevertheless, a limited number of preprints and academic theses were also considered when they provided novel methodologies, datasets, or comparative evaluations that significantly advanced understanding of ATE developments in Arabic. Such inclusions ensured that the review captured recent and emerging research directions, following the broader methodology adopted by Arabic NLP surveys (e.g., [3, 4, 11]).

The rest of the paper is organized as follows: Section 2 discusses the challenges and language-specific issues related to Arabic term extraction. Section 3 reviews the main methods and approaches, including rule-based, statistical, machine learning, deep learning, and hybrid

techniques. Section 4 presents evaluation metrics for assessing Arabic ATE systems. Finally, Section 5 provides conclusions and outlines future research directions.

2. Challenges and Language Specificity

Compared with English, Arabic presents distinctive linguistic challenges for ATE due to its complex morphology, orthographic variation, and syntactic flexibility. English is morphologically simpler and relies heavily on fixed word order, which makes tokenization and part-of-speech tagging more straightforward. In contrast, Arabic words often carry multiple affixes that express tense, gender, number, and definiteness, resulting in high lexical diversity and data sparsity. Additionally, the absence of short vowels (diacritics) in most written Arabic creates orthographic ambiguity, making it difficult for algorithms to distinguish between semantically different terms with identical consonantal roots. Moreover, multiword expressions in Arabic can appear in several syntactic forms due to inflection and agreement rules, while in English they tend to have more stable surface structures. These linguistic differences explain why ATE methods developed for English often require significant adaptation before being effectively applied to Arabic [12] . In this section, we discuss these challenges.

2.1 Complex Morphology

Arabic is morphologically rich, with most words derived roots through various affixes, prefixes, and suffixes. For example, the root k-t-b (كتب) produces terms such as kitab (book, كتاب), maktab (office, مكتب), and katib (writer, كاتب) [8]. Table 1 shows the morphological derivations from the Arabic root k-t-b (كتب). An ATE system must correctly identify and relate these derivations. This requires advanced morphological analysis and normalization techniques.

Table 1: Morphological derivations from the Arabic root k-t-b (كتب)

Root	Term	Transliteration	Meaning	Part of Speech (POS)
كتب	كتاب	*kitāb*	book	Noun
	مكتب	*maktab*	office	Noun
	كاتب	*kātib*	writer	Noun (Agent)

2.2 Orthographic Challenges

Arabic script is written from right to left and often ignores short vowels (diacritics), leading to lexical ambiguity. For example, the consonant sequence ktb (ب , ت , ك) may represent, kutub (books, كُتُب), or kataba (he wrote, كَتَبَ). Without context-sensitive disambiguation, ATE systems extract irrelevant or ambiguous terms [13]. Table 2 shows other examples of Arabic words with same letters but different meanings.

Doi : https://doi.org/10.21608/jaiep.2025.423298.1025
Received: September 13, 2025; Revised: October 13, 2025; Accepted: October 15, 2025

Table 2: Examples of Arabic words with same letters but different meanings

Arabic form	English Meaning	Phonetic transcription form	Transliterations	Part of Speech (POS)
جَد	Grandfather	جَ دُ	Jadd	Noun
جَد	Work hard	جَ دَّ	Jadda	Verb
حُر	Freedom	حُ ر	Ḥurr	Noun
حَر	Hot	حَ ر	Ḥarr	Adjective
حَلْم	Patience	حَ لْ م	Ḥilm	Noun
حِلْم	Forgivingness	حِ لْ م	Ḥilm	Noun
حُلْم	Dream	حُ لْ م	Ḥulm	Noun
عِلْم	Knowledge	عِ لْ م	Elm	Noun
عَلَم	Flag	عَ لَ م	Alam	Noun
عَلِمَ	Knew	عَ لِ مَ	Alima	Verb
عُلِمَ	Is known	عُ لِ مَ	Ulima	Verb
عَلَّمَ	Taught	عَ لَّ مَ	Allama	Verb
عُلِّمَ	Is taught	عُ لِّ مَ	Ullima	Verb

2.3 Syntactic Complexity

Arabic uses a system of case endings that makes word order in sentences more flexible than in many other languages [13]. This flexibility can make it difficult to determine where a term begins and ends. For example, the sentences:

الطالبُ كتبَ الدرسَ(al-ṭalibu kataba al-darsa) – "The student wrote the lesson."
كتبَ الطالبُ الدرسَ(kataba al-ṭalibu al-darsa) – "The student wrote the lesson."

have the same meaning despite the different word order, because the case endings indicate grammatical roles.

In addition, Arabic often omits subject pronouns, as in:

- ذهبتُ (dhahabtu) – "I went," where the pronoun "I" is implied by the verb form.

Similarly, verbs like "is" in English are not always expressed, as in:

- يقرأ الآن (yaqraʾu al-ān) – "He is reading now," without the equivalent of "is" and without explicitly stating the subject pronoun 'He'.

These features make syntactic parsing more challenging. Therefore, effective ATE for Arabic requires parsers that can handle flexible word order and missing elements in a sentence.

2.4 Multiword Expressions (MWEs)

Arabic contains many multiword expressions, including idioms, fixed phrases, and compound nouns. These are groups of words that function as a single meaning unit [14]. For example:

- **يد واحدة لا تصفق** (*yad wāḥida lā tuṣaffiq*) – "One hand cannot clap" (idiom meaning cooperation is necessary).

- **حقوق الإنسان** (*ḥuqūq al-insān*) – "Human rights" (compound noun).

Such expressions can appear in different morphological or syntactic forms. For instance, **حقوق الطفل**(*ḥuqūq al-ṭifl*, "children's rights") shares the same pattern as "human rights" but with a different noun. Idioms may also be slightly rephrased while keeping the same meaning. These variations make it more difficult for ATE systems to identify them as single units.

2.5 Named Entity Recognition (NER) Issues

Named entities in Arabic, including personal names, locations, and organizations, often exhibit multiple forms and spellings [11, 15]. For example: the word **" Center / مركز "** can mean:

- A **named entity** when referring to a specific place, e.g., "مركز القاهرة الدولي" (Cairo International Center).
- A **common term** meaning just "center" or "middle" in any context.

Disambiguating between named entities and common terms is critical but challenging.

2.6 Scarcity of Annotated Data

For Arabic language, high-quality annotated corpora are scarce, making it difficult to train supervised machine learning models [8, 16]. Manual annotation is costly and time-consuming, forcing many systems to rely on small or domain-specific datasets, which limits generalizability.

2.7 Dialectal Variation

Arabic includes many dialects, such as Egyptian, Levantine (Eastern Arabic), Gulf (Khaleeji Arabic), and others, as shown in Fig. 2. Each dialect has its own unique words and sentence structures [17]. For example, the word for "car" in Egyptian Arabic is "عربية" (arabiyya), while in Levantine Arabic it might be "سيارة" (sayyara). Modern Standard Arabic (MSA) is mostly used in formal writing and media. However, online texts often mix MSA with dialects. This mixing can cause problems for automatic extraction systems if the dialectal variations are not addressed properly.

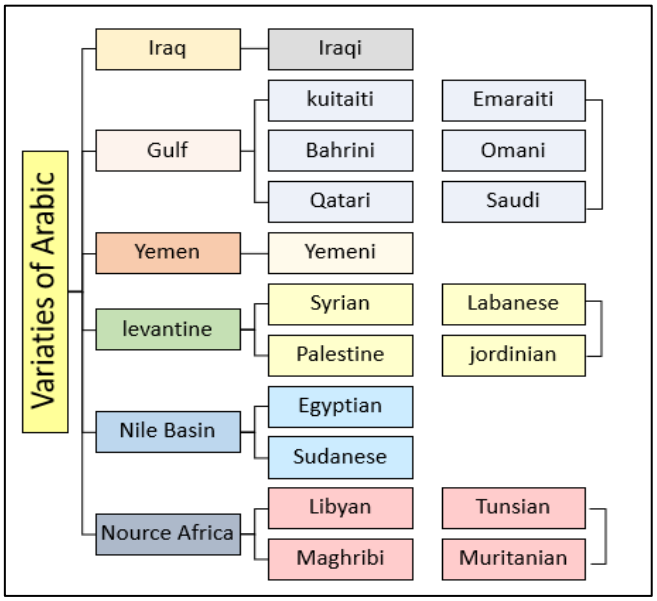

Figure 2: Arabic dialects across countries form a continuous language area

Doi : https://doi.org/10.21608/jaiep.2025.423298.1025
Received: September 13, 2025; Revised: October 13, 2025; Accepted: October 15, 2025

2.8 Domain Adaptation

The performance of ATE systems can vary significantly across domains. For instance, biomedical Arabic corpora often contain highly specialized terms and transliterated words, making automatic extraction more challenging but rewarding for terminology building. In contrast, legal Arabic exhibits rigid syntactic patterns and domain-specific phrasing that may benefit from rule-based or hybrid ATE models. Addressing such differences is essential for designing adaptive, domain-aware ATE systems[18] .

Table 3 shows a summary of key challenges in Arabic Automatic Term Extraction and their impacts. Overcoming these challenges requires a combination of morphological analyzers, context-sensitive disambiguation, robust parsing techniques, and domain-adapted extraction algorithms. These solutions must be integrated into ATE systems specifically designed for Arabic language.

Table 3: Summary of Key Challenges in Arabic Automatic Term Extraction and Their Impacts

Challenge	Impact on ATE	Potential Solutions
Complex Morphology	Missed or incorrect term extraction	Use morphological analyzers and normalization
Orthographic Ambiguities	Ambiguous word interpretation	Apply context-based disambiguation
Syntactic Complexity	Difficult term boundary and MWE detection	Use robust syntactic parsers and sequence models
Multiword Expressions	Fragmented or inaccurate extraction	Detect MWEs with statistical and ML methods
Named Entity Recognition	Confusion between entities and terms	Integrate NER tools and external lexicons
Limited Annotated Data	Poor supervised learning performance	Use semi-supervised or unsupervised learning
Dialectal Variation	Inconsistent extraction across dialects	Focus on MSA; normalize dialects

3. Methods and Approaches

ATE for Arabic text requires specialized methodologies that address the language's unique linguistic characteristics, such as its rich morphology, orthographic ambiguities, and syntactic variability. Over the years, researchers have proposed a variety of approaches, ranging from rule-based systems to modern deep learning models. This section reviews the main categories of methods applied to Arabic ATE, highlighting their principles, strengths, and limitations.

3.1 Rule-Based Approaches

Rule-based methods depend on predefined linguistic rules and resources specifically designed for Arabic. These approaches typically involve:

- **Morphological Analysis:** Tools such as *Buckwalter* and *AraMorph* segment words, identify roots, and handle affixation to normalize variant forms.

- **Dictionary-Based Extraction:** Domain-specific lexicons are matched against the text to identify relevant terms.

- **Part-of-Speech (POS) Tagging:** Arabic POS taggers like *Farasa* and *MADAMIRA* detect nouns and noun phrases, which are prime candidates for term extraction.

The strengths of this approach are its high interpretability and precise control over term selection, while its limitations include labor-intensive rule creation, poor scalability to new domains, and sensitivity to linguistic variations [2].

This approach was widely used in earlier research before other methods emerged, and it has also been combined with other approaches in hybrid systems to overcome its limitations [19–23].

3.2 Statistical and Frequency-Based Methods

These approaches use statistical patterns in text to identify important terms such as:

- **Term Frequency–Inverse Document Frequency (TF–IDF):** Measures the relative importance of a term by comparing its frequency in a document to its frequency in the corpus.

$$TF - IDF(t, d, D) = TF(t, d) \times IDF(t, D) \quad (1)$$

Where:

- $TF(t,d) = \frac{f_{t,d}}{\sum_{t \epsilon d} f_{t,d}}$
  - $f_{t,d} =$ frequency of term t in document d
- $IDF(t,D) = \log(\frac{N}{1+|\{d \in D: t \in d\}|})$
  - $N =$ total number of documents in corpus D
  - $|\{d \in D : t \in d\}| =$ number of documents containing t

- **Pointwise Mutual Information (PMI):** Quantifies word associations to detect potential multiword terms. It Measures the strength of association between two words w_1 *and* w_2.

$$PMI\ (w_1, w_2) = \log\left(\frac{P(w_1, w_2)}{P(w_1).P(w_2)}\right) \quad (2)$$

where:

- $P(w1, w2) =$

  probability of w1 and w2 occuring together (within a window or as a bigram)

- *P(w1).P(w2)* = individual probabilities of w1 and w2

- **Word Co-occurrence Analysis:** Identifies words that frequently occur within a fixed window, suggesting a semantic link.

No single "standard" formula, but often computed as:

$$Co - occurrence(w_i, w_j) = \sum_{windows} 1(w_i \,\&\, w_j\ appear\ together) \quad (3)$$

Or normalized as:

Doi : https://doi.org/10.21608/jaiep.2025.423298.1025
Received: September 13, 2025; Revised: October 13, 2025; Accepted: October 15, 2025

$$Co - occurrence\ \ Score(w_i, w_j) = \frac{count\,(w_i, w_j)}{\sum_{w_k} count\,(w_i, w_k)} \quad (4)$$

where

- $count\,(w_i, w_j) =$

  number of times $w_i\ and\ w_j$ occur together within a defined window size.

- $1(.) =$ indicator function (1 if condition is true, 0 otherwise)

The strengths of these methods are their ease of implementation, domain-independence, and usefulness for identifying frequent collocations, while their limitations include difficulty handling polysemy, inability to capture deep semantic relationships, and lower effectiveness for low-frequency terms.

3.3 Machine Learning Approaches

Machine learning (ML) models learn extraction patterns from data rather than relying solely on rules [5, 11, 24–26].

- **Supervised Learning:** Models such as Support Vector Machines (SVM), Decision Trees, and Conditional Random Fields (CRF) require annotated corpora to learn features like POS tags, word frequency, and syntactic patterns.
- **Unsupervised Learning:** Clustering (e.g., *k*-means) and topic modeling (e.g., LDA) group related words without labeled data.
- **Semi-Supervised Learning:** Combines a small labeled dataset with large amounts of unlabeled data, useful when annotated resources are scarce.

The strengths of these approaches are their adaptability to different domains and their ability to learn complex patterns, while their limitations include the reliance of supervised models on high-quality labeled corpora and the tendency of unsupervised models to produce noisy results.

3.4 Deep Learning Approaches

Deep neural networks capture context and semantics more effectively than traditional ML models [27–30].

- **Recurrent Neural Networks (RNNs) and LSTMs:** Suitable for sequence labeling, they model dependencies between words to detect terms and multiword expressions.
- **Transformer-Based Models (BERT, AraBERT):** Pre-trained contextual embeddings improve term extraction by leveraging bidirectional context and semantic information.
- **Word Embeddings (Word2Vec, fastText):** Represent words in continuous vector space, enabling clustering and similarity-based term detection.

In recent years, Arabic-specific Large Language Models (LLMs) such as AraGPT [31] and Jais [32], along with other transformer-based architectures, have demonstrated remarkable progress in capturing complex linguistic patterns. Their deep contextual understanding and semantic representation indicate strong potential for improving ATE performance, particularly in domain adaptation and in handling morphologically rich Arabic expressions.

The strengths of these approaches are their ability to handle complex dependencies, robustness to linguistic variation, and high performance in contextual term detection, while their

limitations include the need for large training corpora, high computational resources, and lower interpretability compared to rule-based methods.

3.5 Hybrid Approaches

Hybrid systems combine rule-based and statistical or machine learning methods to exploit their complementary strengths. For instance, morphological analyzers can pre-filter candidate terms before applying a classifier, or statistical co-occurrence measures can be combined with deep learning models to refine results [33].

The strengths of these approaches are their often higher accuracy and better handling of Arabic-specific complexities, while their limitations include increased system complexity and higher development effort.

Fig. 3 illustrates the Arabic ATE methods and tools employed in previous approaches.

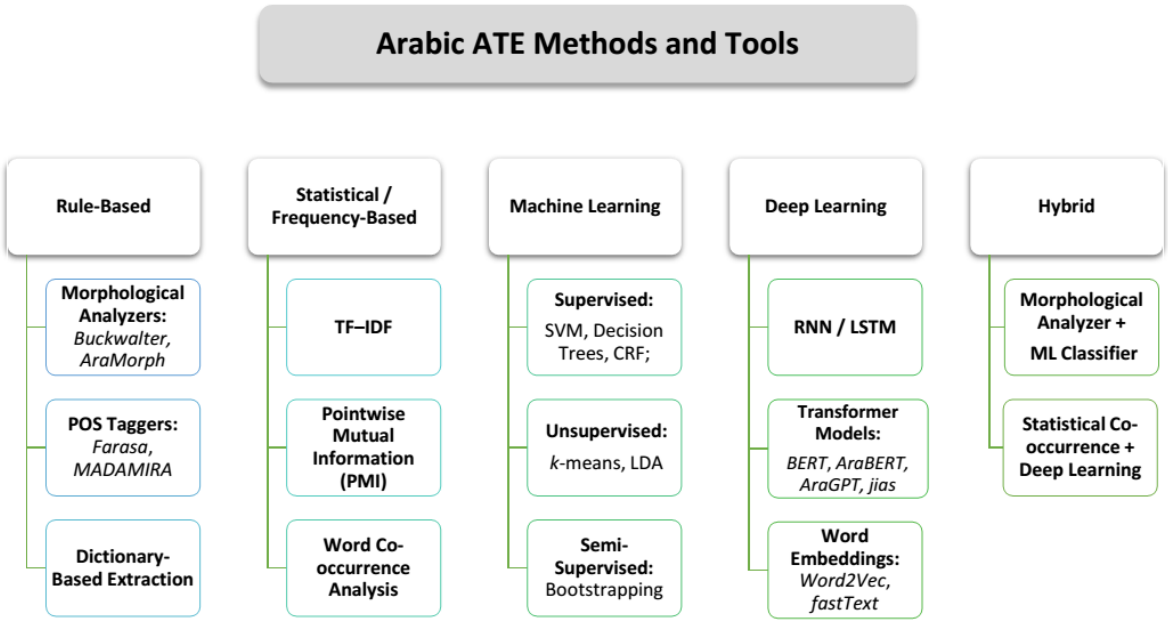

Figure 3: Arabic ATE Methods and Tools

Each ATE approach for Arabic offers distinct trade-offs between precision, scalability, and adaptability. Recent trends show a shift toward deep learning and hybrid systems, which leverage both linguistic expertise and data-driven models to address the complexities of Arabic term extraction [33].

Table 4 compares the different approaches, highlighting their respective advantages and disadvantages.

4. Evaluation Metrics

Evaluating the performance of ATE systems is essential for determining their accuracy, robustness, and applicability to real-world tasks. In the context of Arabic, evaluation must account for the language's morphological richness, syntactic variability, and frequent use of multiword expressions. This section reviews the most widely used metrics in ATE research, emphasizing their relevance and adaptation to Arabic [9, 10, 14].

Doi : https://doi.org/10.21608/jaiep.2025.423298.1025
Received: September 13, 2025; Revised: October 13, 2025; Accepted: October 15, 2025

Table 4: Advantages and Disadvantages of ATE approaches

Approach	Advantages	Disadvantages
Rule-Based	High interpretability; precise control; effective for well-defined domains.	Labor-intensive rule creation; poor scalability; sensitive to variation.
Statistical / Frequency-Based	Easy to implement; domain-independent; good for frequent collocations.	Struggles with polysemy; misses low-frequency terms; shallow semantics.
Machine Learning	Adaptable; learns complex patterns; supports multiple domains.	Requires quality labeled data (for supervised); noisy results (unsupervised).
Deep Learning	Captures context and semantics; robust to variation; high accuracy.	Needs large corpora; high computational cost; less interpretable.
Hybrid	Combines strengths of multiple methods; higher accuracy for complex cases.	Complex development; integration overhead; higher maintenance needs.

4.1 Precision

Precision measures the proportion of extracted terms that are correct according to a gold standard as in Eq. (5):

$$Precision = \frac{Number\ of\ correct\ terms\ extracted}{Total\ number\ of\ terms\ extracted} \quad (5)$$

High precision indicates that the system avoids extracting irrelevant or incorrect terms critical for Arabic due to ambiguity and polysemy, for example: If 100 terms are extracted and 80 are correct, precision is 80%.

4.2 Recall

Recall measures the proportion of relevant terms in the corpus that the system successfully extracts as in Eq. (6):

$$Recall = \frac{Number\ of\ correct\ terms\ extracted}{Total\ number\ of\ relevant\ terms\ in\ corpus} \quad (6)$$

For Arabic ATE, high recall ensures that morphological variants, multiword expressions, and less frequent terms are captured, for example: If 120 relevant terms exist and the system finds 80, recall is 0.67 (67%).

4.3 F1-Score

The F1-score is the harmonic mean of precision and recall, providing a balanced measure of system performance as in Eq. (7):

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

In Arabic ATE, where precision and recall often trade off, the F1-score offers a single metric that captures both.

4.4 Multiword Expression (MWE) Recognition Accuracy

Given Arabic's frequent use of MWEs, this metric measures how accurately a system identifies multiword terms as cohesive units rather than separate words as in Eq. (8).

$$MWE\ Accuracy = \frac{Number\ of\ correct\ MWEs\ extracted}{Total\ number\ of\ MWEs\ in\ gold\ standard} \tag{8}$$

4.5 Corpus-Based Evaluation

In this method, system outputs are compared to a manually annotated reference corpus. This approach is highly reliable for domain-specific ATE in Arabic but depends heavily on the quality and representativeness of the corpus.

Table 5 summarizes the advantages and limitations of each ATE metric. It provides a clear comparison, highlighting the strengths of different metrics in handling term extraction tasks, as well as their potential drawbacks. This overview helps in selecting the most appropriate metric for a given application.

Table 5: Advantages and Limitations of ATE Metrics

Metric	Advantages	Limitations
Precision	Measures extraction accuracy; easy to interpret	Ignores missed terms; may favor conservative systems
Recall	Measures coverage; ensures relevant terms are found	May favor overly inclusive systems with low precision
F1-Score	Balances precision and recall into one metric	Cannot distinguish causes of low score
MWE Recognition Accuracy	Captures ability to extract cohesive multiword terms	Requires high-quality MWE annotations
Corpus-Based Evaluation	Provides highly reliable benchmark	Requires costly and time-consuming annotation

5. Conclusion and Future Work

Automatic Term Extraction (ATE) is an important step in many Arabic NLP applications, but it faces special challenges because of Arabic's complex word structures, spelling variations, flexible grammar, and frequent use of multiword expressions. In this paper, we reviewed

Doi : https://doi.org/10.21608/jaiep.2025.423298.1025
Received: September 13, 2025; Revised: October 13, 2025; Accepted: October 15, 2025

different approaches including rule-based, statistical, machine learning, deep learning, and hybrid methods. Each of these has strengths and weaknesses: while rule-based and statistical methods are simple and clear, they struggle with scalability and deep meaning; machine learning and deep learning models are more powerful but need large datasets; and hybrid methods often give the best balance. We also highlighted evaluation metrics that extend beyond precision and recall to capture the unique linguistic difficulties of Arabic, such as detecting correct word boundaries and multiword expressions.

Although progress has been made, more work is still needed to improve Arabic ATE. Future research should create larger and better annotated datasets, design systems that can deal with both Modern Standard Arabic and dialects, and develop models that adapt easily across different domains. Linking extracted terms with knowledge resources like ontologies and WordNet will also improve their usefulness. In addition, lighter and faster models are important for practical applications. Advancing in these directions will help build more accurate, efficient, and widely usable ATE systems for Arabic texts.

References

1. A. M. Al-Thubaity, M. Khan, S. Alotaibi, and B. Alonazi, "Automatic Arabic term extraction from special domain corpora," in *2014 international conference on Asian language processing (IALP)*, IEEE, 2014, pp. 1–5.
2. T. Wissik, "Impact of automatic term extraction on terminology work: A qualitative interview study in institutional settings," *Terminology*, vol. 31, no. 1, pp. 110–135, 2025.
3. K. Al Khatib and A. Badarneh, "Automatic extraction of Arabic multi-word terms," in *Proceedings of the International Multiconference on Computer Science and Information Technology*, IEEE, 2010, pp. 411–418.
4. K. Darwish, N. Habash, M. Abbas, H. Al-Khalifa, H. T. Al-Natsheh, H. Bouamor, K. Bouzoubaa, V. Cavalli-Sforza, S. R. El-Beltagy, W. El-Hajj, and M. Jarrar, "A panoramic survey of natural language processing in the Arab world," *Commun ACM*, vol. 64, no. 4, pp. 72–81, 2021.
5. S. L. Marie-Sainte, N. Alalyani, S. Alotaibi, S. Ghouzali, and I. Abunadi, "Arabic natural language processing and machine learning-based systems," *IEEE Access*, vol. 7, pp. 7011–7020, 2018.
6. S. AbdelRahman, M. Elarnaoty, M. Magdy, and A. Fahmy, "Integrated machine learning techniques for Arabic named entity recognition," *IJCSI*, vol. 7, no. 4, pp. 27–36, 2010.
7. A. Benabdallah, M. A. Abderrahim, and M. E.-A. Abderrahim, "Extraction of terms and semantic relationships from Arabic texts for automatic construction of an ontology," *Int J Speech Technol*, vol. 20, no. 2, pp. 289–296, 2017.
8. W. Lahbib, I. Bounhas, and B. Elayeb, "Arabic-English domain terminology extraction from aligned corpora," in *On the Move to Meaningful Internet Systems: OTM 2014 Conferences: Confederated International Conferences: CoopIS, and ODBASE 2014, Amantea, Italy, October 27-31, 2014, Proceedings*, Springer, 2014, pp. 745–759.
9. Y. Jaafar and K. Bouzoubaa, "A survey and comparative study of Arabic NLP architectures," *Intelligent Natural Language Processing: Trends and Applications*, pp. 585–610, 2018.
10. F. Enríquez, F. L. Cruz, F. J. Ortega, C. G. Vallejo, and J. A. Troyano, "A comparative study of classifier combination applied to NLP tasks," *Information Fusion*, vol. 14, no. 3, pp. 255–267, 2013.
11. G. Petasis, F. Vichot, F. Wolinski, G. Paliouras, V. Karkaletsis, and C. D. Spyropoulos, "Using machine learning to maintain rule-based named-entity recognition and classification systems," in *proceedings of the 39th annual meeting of the association for computational linguistics*, 2001, pp. 426–433.
12. A. El Mahdaouy, S. E. L. Ouatik, and E. Gaussier, "A study of association measures and their combination for Arabic MWT extraction," *arXiv preprint arXiv:1409.3005*, 2014.
13. M. Khader, A. Awajan, and A. Alkouz, "Textual entailment for Arabic language based on lexical and semantic matching," *International Journal of Computing & Information Sciences*, vol. 12, no. 1, pp. 67–74, 2016.

14. M. Attia, A. Toral, L. Tounsi, P. Pecina, and J. Van Genabith, "Automatic extraction of Arabic multiword expressions," in *Proceedings of the 2010 workshop on multiword expressions: From theory to applications*, 2010, pp. 19–27.
15. M. N. A. Ali, G. Tan, and A. Hussain, "Boosting Arabic named-entity recognition with multi-attention layer," *IEEE Access*, vol. 7, pp. 46575–46582, 2019.
16. A. Šajatović, M. Buljan, J. Šnajder, and B. D. Bašić, "Evaluating automatic term extraction methods on individual documents," in *Proceedings of the Joint Workshop on Multiword Expressions and WordNet (MWE-WN 2019)*, 2019, pp. 149–154.
17. A. Elnagar, S. M. Yagi, A. B. Nassif, I. Shahin, and S. A. Salloum, "Systematic literature review of dialectal Arabic: identification and detection," *IEEE Access*, vol. 9, pp. 31010–31042, 2021.
18. A. Al-Thubaity, "A Novel Dataset for Arabic Domain Specific Term Extraction and Comparative Evaluation of BERT-Based Models for Arabic Term Extraction," *ACM Transactions on Asian and Low-Resource Language Information Processing*.
19. R. Alfred, L. C. Leong, C. K. On, and P. Anthony, "Malay named entity recognition based on rule-based approach," 2014.
20. D. N. Shah and H. B. Bhadka, "Named entity recognition from Gujarati text using rule-based approach," in *Intelligent Systems Design and Applications: 17th International Conference on Intelligent Systems Design and Applications (ISDA 2017) held in Delhi, India, December 14-16, 2017*, Springer, 2018, pp. 797–805.
21. R. Salah, M. Mukred, L. Qadri binti Zakaria, R. Ahmed, and H. Sari, "A new rule-based approach for classical Arabic in natural language processing," *Journal of Mathematics*, vol. 2022, pp. 1–20, 2022.
22. F. Harrag, A. Al-Nasser, A. Al-Musnad, R. Al-Shaya, and A. S. Al-Salman, "Using association rules for ontology extraction from a Quran corpus," in *Proc. 5th Int. Conf. Arabic Language Process*, 2014, pp. 1–8.
23. A. Elsebai, F. Meziane, and F. Z. Belkredim, "A rule based persons names Arabic extraction system," *Communications of the IBIMA*, vol. 11, no. 6, pp. 53–59, 2009.
24. S. AbdelRahman, M. Elarnaoty, M. Magdy, and A. Fahmy, "Integrated machine learning techniques for Arabic named entity recognition," *IJCSI*, vol. 7, no. 4, pp. 27–36, 2010.
25. B. Armouty and S. Tedmori, "Automated keyword extraction using support vector machine from Arabic news documents," in *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*, IEEE, 2019, pp. 342–346.
26. B. Sulistio, A. Ramadhan, E. Abdurachman, M. Zarlis, and A. Trisetyarso, "The utilization of machine learning on studying Hadith in Islam: A systematic literature review," *Educ Inf Technol (Dordr)*, pp. 1–39, 2023.
27. M. M. A. Najeeb, "Towards a deep leaning-based approach for hadith classification," *European Journal of Engineering and Technology Research*, vol. 6, no. 3, pp. 9–15, 2021.
28. M. Al-Smadi, S. Al-Zboon, Y. Jararweh, and P. Juola, "Transfer learning for Arabic named entity recognition with deep neural networks," *IEEE Access*, vol. 8, pp. 37736–37745, 2020.
29. P. Goyal, S. Pandey, and K. Jain, *Deep Learning for Natural Language Processing: Creating Neural Networks with Python*, 1st ed. Berkeley, CA: Apress, 2018.
30. N. Alsaaran and M. Alrabiah, "Classical Arabic named entity recognition using variant deep neural network architectures and BERT," *IEEE Access*, vol. 9, pp. 91537–91547, 2021.
31. W. Antoun, F. Baly, and H. Hajj, "AraGPT2: Pre-trained transformer for Arabic language generation," *arXiv preprint arXiv:2012.15520*, 2020.
32. N. Sengupta, S. K. Sahu, B. Jia, S. Katipomu, H. Li, F. Koto, W. Marshall, G. Gosal, C. Liu, and Z. Chen, "Jais and jais-chat: Arabic-centric foundation and instruction-tuned open generative large language models," arXiv preprint arXiv:2308.16149, 2023.
33. M. Hadni, A. Lachkar, and S. A. Ouatik, "Multi-Word Term Extraction based on New Hybrid Approach for Arabic Language," in CS & IT Conference Proceedings, CS & IT Conference Proceedings, 2014.

Doi : https://doi.org/10.21608/jaiep.2025.423298.1025
Received: September 13, 2025; Revised: October 13, 2025; Accepted: October 15, 2025

Appendix A. Comparison between Arabic and English in ATE

Feature	Arabic	English	Impact on ATE	Example
Morphology	Root-and-pattern morphology generates many derived and inflected forms from a single root.	Mostly linear morphology with limited inflection.	Makes it difficult to recognize all term variants related to one concept.	Arabic: Root ك ت ب produces كاتب (writer), مكتوب (written), كتاب (book). English, write → writer, writing (fewer variations).
Orthography	Short vowels are often omitted in writing, causing ambiguity.	All vowels are written explicitly.	Increases term ambiguity and affects token matching.	Arabic: علم can mean ʿilm (knowledge) or ʿalam (flag). English, "flag" has only one orthographic form.
Tokenization	Words may include attached clitics (e.g., conjunctions, prepositions, pronouns).	Words are usually separated by spaces.	Makes tokenization and term boundary detection more complex.	وبالمدرسة = و + ب + المدرسة ("and + in + the school"). English equivalent: "in the school."
Multiword Expressions (MWEs)	Very frequent and semantically rich, often metaphorical or idiomatic.	Also common but more straightforward morphologically.	Requires semantic and contextual analysis to extract accurately.	حقوق الإنسان (human rights), بيت الشعر (tent or "house of poetry" depending on context).
Orthographic Variation	Multiple valid spellings for the same word (with/without Hamza, Ta Marbuta, Alif Maqsura).	Consistent spelling system.	Causes difficulty in normalization and term matching.	مسؤول vs. مسئول (both mean "responsible").
Named Entities	May appear without capitalization, making them hard to detect.	Capital letters signal proper nouns.	Reduces accuracy of named-entity-based term extraction.	Arabic: مصر (Egypt) looks like a regular noun; English: Egypt is clearly marked as a name.