# Unsupervised Keyphrase Prediction: Methods and Evaluation

Huiqian Wu and Yuqing Sun(✉)

Shandong University, Jinan 250000, China
sun_yuqing@sdu.edu.cn

**Abstract.** Keyphrases represent the core content of a document, facilitating efficient information processing in knowledge discovery systems. Traditional supervised keyphrase prediction methods not only rely on labeled data but also lack robustness across diverse domains. Recently, with the advancements in pretrained language models, unsupervised methods have gained increasing attention. This survey provides a comprehensive review of the entire process of unsupervised keyphrase prediction. We begin with an analysis of the linguistic properties of keyphrases, aiming to support the design and evaluation of keyphrase prediction models. Then, we categorize and discuss the details of existing unsupervised methods for both keyphrase extraction and generation, emphasizing cutting-edge techniques such as attention mechanisms and prompt learning. Additionally, we examine evaluation metrics, introduce a novel reference-free metric, and provide a list of open-source datasets. Finally, we explore promising future directions and conclude the survey.

**Keywords:** Unsupervised Keyphrase prediction · Keyphrase extraction · Keyphrase generation · Keyphrase evaluation

## 1 Introduction

Keyphrase prediction(KP) aims to identify a set of phrases from unstructured documents that encapsulate the core semantic information of the documents. The target elements include present keyphrases, which appear in the source document [32], as well as absent keyphrases, which are semantically related to but exhibit no lexical overlap or only partial overlap with the source document [30] (Fig. 1). The keyphrase prediction task is categorized into two types: 1)Keyphrase Extraction(KPE): Focuses on extracting only present keyphrases; 2)Keyphrase Generation(KPG): Generates both present and absent keyphrases. Keyphrases play a critical role as information-dense units in supporting downstream tasks such as information retrieval [7].

Traditional keyphrase extraction methods treat a document as a bag of words and extract present keyphrases based on expert-defined rules [9]. Graph-based methods incorporate word relations to obtain the relevance between phrases and documents. Supervised neural network models can capture the implicit semantics of documents, which enhances keyphrase prediction.

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2025
X. Wu et al. (Eds.): PAKDD 2025, LNAI 15873, pp. 417–434, 2025.
https://doi.org/10.1007/978-981-96-8183-9_30

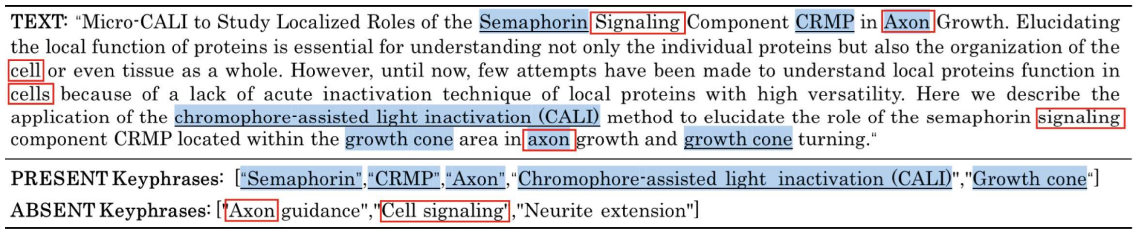


**Fig. 1.** An example from Kp-Biomed [17] dataset. The present keyphrases are highlighted in blue, and the overlapping parts of the absent keyphrases with the source document are outlined in red. (Color figure online)

However, those supervised models rely on high-cost labeled data and perform suboptimally in out-of-domain documents [12]. Therefore, unsupervised keyphrase prediction, which offers better flexibility and domain generalization, have been extensively studied.

Recent advances in unsupervised keyphrase prediction have focused on leveraging pretrained language models(PLMs), particularly large language models. They possess strong text understanding and generation capabilities, demonstrating impressive performance in zero-shot NLP tasks. Cutting-edge techniques based on PLMs, such as prompt learning, have significantly enhanced the performance of unsupervised keyphrase prediction. Nonetheless, challenges still remain, including the tendency to generate repetitive keyphrases and the difficulty in producing absent keyphrases.

To address these challenges and advance the optimization of unsupervised keyphrase prediction models, this survey provides a systematic analysis of the task . The primary contributions are summarized as follows:

**1. Analysis of Keyphrase Properties.** We analyze the intrinsic linguistic properties of keyphrases, providing insights for model design and evaluation.

**2. Taxonomy of Unsupervised Keyphrase Prediction Methods.** We categorize unsupervised keyphrase extraction and generation methods, with a focus on state-of-the-art representative techniques.

**3. Evaluation Metrics: Discussion and Proposal.** We comprehensively review existing evaluation metrics and introduce a novel reference-free metric designed to enhance evaluation robustness.

The remainder of this paper is organized as follows: Sect. 2 emphasizes the properties of keyphrases and provides an overview of the keyphrase prediction task. Sects. 3 and 4 introduce unsupervised methods for keyphrase extraction and generation, respectively. Sect. 5 explores current evaluation metrics, proposes new reference-free metrics, and enumerates relevant datasets. Sect. 6 discusses future directions for unsupervised keyphrase prediction. Sect. 7 concludes the survey.

## 2 Overview

### 2.1 Keyphrase Property Analysis from a Linguistic Perspective

To guide the design and evaluation of keyphrase prediction models, we analyze the syntactic and semantic properties of keyphrases from a linguistic perspective.

**Syntactic Properties.** Syntactic properties refer to the structural characteristics that keyphrases should have in terms of their grammatical form. Firstly, keyphrases can be categorized into present and absent keyphrases based on whether their constituent words appear consecutively in the source text(Fig. 1). According to statistics from [30], present keyphrases constitute 55.69% of the Inspec dataset [18], 44.74% of the Krapivin2009 dataset [24], 67.75% of the NUS dataset [33], and 42.01% of the SemEval dataset [22], with the remaining keyphrases being absent. Further research [6] redefined absent keyphrases, replacing the binary classification with a four-class system. Building on this, the Kp-Biomed biomedical dataset was developed [17].

Secondly, keyphrases have no fixed length or boundary restrictions and are typically composed of zero or more adjectives followed by one or more nouns. They should preserve the original word forms as in source documents, avoiding unnecessary length or redundancy. Therefore, we define the first property of a keyphrase as conciseness. ***Conciseness*** *means that a keyphrase should be a noun phrase reflecting the content of the source document without adding or omitting boundary words.* For example, in Fig. 1, "Chromophore-assisted light inactivation" becomes unnecessarily long when extended to "Chromophore-assisted light inactivation method", as "method" does not add thematic relevance. Conversely, shortening it to "light inactivation" omits the critical modifier "Chromophore-assisted", failing to capture the original meaning.

**Semantic Properties.** Semantic properties refer to the meaning characteristics that keyphrases should possess. Based on previous studies [13,52], a keyphrase collection should summarize document essentials and ensure inter-document distinctiveness. This contrasts with the "high-quality phrases" in phrase mining, which focuses more on domain popularity.

From the perspective of the relationship between keyphrases and the source document, we define the second property of a keyphrase set as coverage. ***Coverage*** *refers to the extent to which the keyphrase set captures the document's topics while preserving critical information.* While information loss is inevitable when transforming a long document into keyphrases, a good keyphrase set minimizes this loss. Additionally, coverage encourages the appearence of absent keyphrases, which can better summarize the entire information. For example, in Fig. 1, the absent keyphrase "Neurite extension" succinctly encapsulates the main idea of the entire abstract at a high level.

Next, from the perspective of the relationship between keyphrases and the domain, we define the third property of a keyphrase as distinctiveness. ***Distinctiveness*** *refers to a keyphrase's ability to distinguish the current document from others, whether within the same domain or across different domains.* Phrases commonly used across multiple domains lack distinctiveness. For instance, words like "protein" and "cell" are prevalent in the biomedical domain, whereas more specific phrases like "cell signaling" are less common and can better differentiate the source document (Fig. 1). As keyphrase prediction is increasingly applied to

domain-specific texts, distinctiveness ensures that keyphrases are more effective in supporting the analysis of specialized content.

### 2.2 Taxonomies of Unsupervised Keyphrase Prediction Methods

This survey comprehensively analyze the recent models on unsupervised keyphrase prediction, with a particular emphasis on cutting-edge techniques involving pretrained language models. As illustrated in Table 1, unsupervised keyphrase prediction methods are systematically divided into two principal categories: unsupervised keyphrase extraction (UKPE, Sect. 3) and unsupervised keyphrase generation (UKPG, Sect. 4). The discussion of methods follows the chronological development of keyphrase prediction technology.

**Table 1.** The taxonomy of representative studies on unsupervised keyphrase prediction

| Category | | Methods |
|---|---|---|
| **Extraction** | | |
| **Traditional** | Syntactic Feature | PoS-tagging 2003 [18], YAKE 2020 [9], HAKE 2022 [31] |
| | Statistical Feature | TF-IDF 1972 [45], YAKE 2020 [9], HAKE 2022 [31] |
| **Graph-based** | Expanded Text | SingleRank 2008 [47], CiteTextRank 2014 [16] |
| | Topic | TopicRank 2013 [8], MultipartiteRank 2018 [4] |
| | Position | PositionRank 2017 [14], JointGL 2021 [27] |
| | Embedding | JointGL 2021 [27] |
| **PLM-based** | Embedding Similarity | EmbedRank 2018 [3], SIFRank 2020 [46]; MDERank 2021 [55], HGUKE 2023 [42]; CentralityRank 2023 [44], HGRRM 2023 [57], HyperRank 2023 [43], INSPECT 2023 [19]; KPEBERT 2021 [55], KeyBART 2021 [25] |
| | Attention Mechanism | AttentionRank 2021 [11], SAMRank 2023 [20] |
| | Prompt Learning | PromptRank 2023 [23], KPE-prompt Comparison 2024 [39]; ChatGPT-prompting 2023 [40] |
| **Generation** | | |
| **Pseudo Data-based** | From Datasets | AutoKeyGen 2022 [38], TPG 2024 [21], Silk 2024 [5] |
| | From Corpora | OpenDomainKP 2023 [12] |
| **LLM-based** | Prompt Learning | LLMs Prompt Learning 2024 [41,48], Long Document Processing 2023 [29], Output Correction 2024 [51] |

* Note: PLM = Pretrained Language Models; LLM = Large Language Models.

# 3 Unsupervised Keyphrase Extraction Methods

## 3.1 Traditional Unsupervised Keyphrase Extraction

Traditional unsupervised keyphrase extraction models rely on word-level features of the source document, primarily focusing on **syntactic** and **statistical** information to design matching rules. We introduce them respectively below.

**1) Syntactic Feature-based Methods.** Keyphrases are typically noun phrases, composed of one or more nouns, which may be preceded by one or more adjectives. They can be represented by the part-of-speech pattern <NN.|JJ> <NN.*>, where "NN" denotes nouns and "JJ" denotes adjectives. The first step of most unsupervised methods is to match candidate phrases from the document according to this pattern [18]. The process often involves tokenizing the text, tagging parts of speech using StanfordCoreNLP tools[1], and selecting phrases that match the pattern using NLTK[2]. Researchers have also introduced various syntactic rules and leveraged morphological features such as phrase length, upper cases, and abbreviations [9] to minimize the influence of irrelevant terms.

**2) Statistical Feature-based Methods.** Words that frequently appear in a document but rarely in the corpus are considered more relevant to the document and thus have higher TF-IDF scores [45] . As a fundamental and effective statistical measure, TF-IDF is commonly integrated into unsupervised keyphrase extraction methods [55]. For example, the AutoKeyGen model [38] uses TF-IDF to generate pseudo-labels. Additionally, co-occurrence relationships can capture the contextual meaning of words by considering their frequent pairings within the corpus, assisting in determining the importance of phrases.

Many traditional keyphrase extraction approaches combine both syntactic and statistical features, such as YAKE [9] and HAKE [31]. These methods integrate part-of-speech patterns, word frequency, and other features to select keyphrases. By leveraging multiple features, these hybrid methods can capture the phrase importance in different dimensions, enabling efficient and domain-agnostic keyphrase extraction with solid performance.

## 3.2 Graph-Based Unsupervised Keyphrase Extraction

Graph-based unsupervised keyphrase extraction methods determine the importance of words by modeling their relationships in a graph, where nodes represent words and edges denote co-occurrence within a specified window. Later developments have been made by incorporating the **expanded text**, **topic**, **position**, and **embedding** information to enrich the attributes of nodes and edges. We introduce these four aspects respectively below.

**1) Expanded Text-based Methods.** Keyphrase sets from similar texts often overlap, allowing models to leverage these similarities to enhance the original graph's nodes and edges, highlighting important terms. For instance, the SingleRank model [47] queries similar documents and builds a graph over the

[1] https://stanfordnlp.github.io/CoreNLP/.
[2] https://github.com/nltk.

expanded text set. Similarly, the CiteTextRank model [16] utilizes the citation network of scientific papers to expand the context and constructs a graph based on the vocabulary found within the citation context.

**2) Topic-based Methods.** Documents typically encompass a central theme and multiple sub-themes, each represented by different keyphrases. Assigning keyphrases to distinct topics helps prevent the keyphrase set from becoming homogeneous. The TopicRank model [8] clusters phrases into topics and selects the most representative phrase from each topic. The MultipartiteRank model [4] extends TopicRank by using keyphrases as nodes and creating directed edges between nodes from different topics, with no edges within the same topic.

**3) Position-based Methods.** Text boundaries, such as titles and paragraph beginnings and endings, often contain key content. Therefore, the PositionRank model [14] prioritizes words appearing at the beginning of the document. The JointGL model [27] obtains phrase positions using a boundary function $d_b(i) = \min(i, \alpha(n-i))$, where n is the total number of candidate keyphrases and $\alpha$ is a hyperparameter controlling the importance of the document boundaries . If $d_b(i) < d_b(j)$, phrase $i$ is closer to the boundary. Then the JointGL model reduces the centrality contribution of $j$ to $i$, making sure that phrases at the start or the end of a document are considered more important than others.

**4) Embedding-based Methods.** Pretrained word embeddings capture semantic information that can enrich the relationships in a graph. Compared to co-occurrence relationships, embedding similarity captures semantic relationships between nodes more effectively, leading to improved performance in unsupervised keyphrase extraction. For instance, the JointGL model [27] employs BERT [10] to encode nodes, with edge weights determined by the dot product of their embeddings. Advances in pretrained language models have made graph construction based on semantic embedding similarity as the mainstream approach.

### 3.3 Pretrained Language Model-Based Unsupervised Keyphrase Extraction

Pretrained language models possess extensive linguistic knowledge and strong semantic understanding, making them highly effective for keyphrase extraction. As a result, methods based on these language models have gained significant attention in recent research, representing the cutting-edge technology. These approaches often incorporate techniques such as **embedding similarity computation**, **attention mechanisms**, and **prompt learning**. Below, we provide an overview of these three key techniques.

**Embedding Similarity-based Methods.** Pretrained language models encode the semantic meaning of text as low-dimensional vectors, which can be used for similarity calculation. The EmbedRank model [3] uses PLMs to encode the text and candidate keyphrases and ranks keyphrases by computing cosine similarity. With continuous optimization of pretrained language models, semantic representation has been significantly improved. For instance, the SIFRank model [46]

enhances the embeddings in EmbedRank with stronger PLMs. Subsequent models have been continuously refined from various perspectives, including long document processing, hierarchical semantic modeling, and task-specific pretraining. These advancements are discussed below.

**1) Long Document Processing.** When processing long documents, the disparity in sequence lengths between candidate keyphrases and the document can degrade the accuracy of embedding similarity calculations. To address this issue, the MDERank model [55] replaces the candidate keyphrases in the source document with the [MASK] token to create a masked document, then calculates the embedding similarity between the masked document and the source document, indirectly ranking the candidate keyphrases. The HGUKE model [42] uses a subset of the source document as a proxy for the entire text, reducing the influence of general content and emphasizing key information.

**2) Hierarchical Semantic Modeling.** Directly calculating the embedding similarity between candidate phrases and the document overlooks the multi-level semantic information within the document, often resulting in a homogeneous keyphrase set. To address this, the CentralityRank model [44] computes embeddings at three levels (word, phrase, and document) and calculates the relevance of each candidate phrase to these levels, ranking them accordingly. Similarly, the HGRRM model [57] assesses sentence importance before ranking keyphrases within each sentence. The HyperRank model [43] captures hierarchical semantic information in hyperbolic space, facilitating richer tree-like representations. It maps both phrase embeddings and document embeddings to the same hyperbolic space and then calculates their Poincaré distance, effectively capturing semantic proximity.

Instead of explicitly modeling multi-level semantics at the phrase, sentence, and document levels, the INSPECT model [19] implicitly captures the topic information of the source document. It assigns keyphrases to distinct topics, thereby maintaining the diversity of the keyphrase set.

**3) Task-specific Pretraining.** To improve keyphrase embedding learning, researchers have proposed task-specific self-supervised pretraining tasks. KPEBERT [55] is further pretrained on BERT with contrastive learning between the original documents, document masked important phrases, and documents masked with general phrases. Similarly, KeyBART [25] employs multi-task pretraining, including random token masking, keyphrase boundary filling, and keyphrase replacement classification. KeyBART significantly outperforms the original model across multiple tasks, including keyphrase extraction and named entity recognition.

**Attention Mechanism-based Methods.** The attention mechanism assigns context-aware scores to each word position for a given query, capturing their semantic relevance within the text. The fixed key-query-value (KQV) matrices in pretrained language models enable direct deployment without retraining. By leveraging the attention mechanism, unsupervised keyphrase extraction models offer both simplicity and state-of-the-art performance across diverse domains.

The AttentionRank model [11] uses BERT to calculate the self-attention scores and cross-attention scores. The self-attention score $a_c$ measures the relevance of the candidate phrase $c$ to the original sentence, while cross-attention improves the embedding similarity score $r_c$ between the phrase and the document. The final ranking score for candidate phrases is computed as a linear combination of $a_c$ and $r_c$. Similarly, the SAMRank model [20] defines the importance score of a phrase by combining global attention and proportional attention scores. First, it extracts the self-attention matrix from models like BERT or GPT-2 to calculate the global attention score of a phrase, which is the sum of attention values from all other tokens to the phrase. Second, SAMRank calculates the proportional attention score by observing that a token attending strongly to an important token is itself important. The final importance score for a phrase is the sum of its global and proportional attention scores.

Building on the attention mechanism, leveraging a larger and more advanced language model enables more precise attention score computations, leading to enhanced task performance.

**Prompt Learning-based Methods.** Prompt learning utilizes natural language prompts to activate the knowledge embedded in pretrained language models. Task-specific prompt templates can be designed for unsupervised keyphrase extraction, applicable to both lightweight pretrained models and large language models.

**1) Prompt Learning on PLMs.** For instance, the PromptRank model [23] employs a pretrained language model as its backbone and constructs prompt templates such as *"Book: [Document]"* and *"This book mainly discusses [Candidate Phrase]"*, where the document and the candidate phrase are dynamically inserted. These prompts are then encoded into a shared latent space. The relevance of between the candidate phrase and the document is determined by the probability of generating $c$. The probability is calculated in Eq. 1, where $j$ is the starting position of the candidate phrase, $l_c$ is the length of the candidate phrase, and $\alpha$ is a hyperparameter to control the model's bias towards shorter or longer phrases. Additionally, based on the assumption that important information often appears at the beginning of the document, PromptRank introduces a position penalty $r_c$. The final score for the candidate phrase is then computed as $s_c = r_c \times p_c$.

$$p_c = \frac{1}{(l_c)^\alpha} \sum_{i=j}^{j+l_c-1} \log p(y_i | y_{<} i) \tag{1}$$

Although PromptRank demonstrates strong performance with manually crafted prompts, designing effective prompts necessitates domain expertise and rigorous experimentation, highlighting the inherent challenges of prompt engineering in unsupervised keyphrase extraction. A study [39] investigated the effect of prompt complexity on keyphrase extraction across six benchmark datasets and multiple pretrained models, demonstrating that simpler prompts can match or exceed the performance of more complex ones.

**2) Prompt Learning on LLMs.** Large language models with increased parameter sizes have recently exhibited remarkable performance in various zero-

shot natural language processing tasks [36]. A recent study [40] tested various natural language prompts using ChatGPT [34] for keyphrase extraction. The study found that while large language models excel in zero-shot scenarios, they underperform state-of-the-art unsupervised keyphrase extraction methods on standard automatic evaluation metrics. Moreover, the structure and phrasing of prompts significantly influence the extraction results. Future research should focus on designing more effective prompts to fully harness the potential of large language models for unsupervised keyphrase extraction.

## 4 Unsupervised Keyphrase Generation Methods

### 4.1 Pseudo Data-based Unsupervised Keyphrase Generation

Supervised neural network models have achieved state-of-the-art performance on keyphrase generation tasks. However, the scarcity of high-quality labeled data makes synthetic data a viable alternative for training. Synthetic pseudo data is typically derived from two sources: existing datasets and external corpora. The following section outlines the two primary ways for generating such data.

**From Datasets.** Texts from the same dataset typically belong to a common domain, such as the KP20k [30] dataset for computer science, leading to overlapping keyphrase sets across documents. Thus, the AutoKeyGen model [38] uses KP20k to generate pseudo-labels. It first adds noun phrases from the dataset into a phrase bank, then obtains the candidate set based on their occurrence in the input document. Then, the ranked candidate phrases are chosen as pseudo-label data to train a sequence-to-sequence model.

However, directly constructing the phrase bank from datasets neglects the varying importance of different textual components. To address this limitation, a recent study [21] generates pseudo-labels from document titles, enhancing absent keyphrase generation by prioritizing title significance. Similarly, the Silk model [5] generates pseudo-labels from citation contexts by applying principles of importance, relevance, and reliability, achieving robust performance across domains such as natural language processing, astrophysics, and paleontology. However, it relies on domain-specific datasets to construct effective pseudo-labels for diverse fields.

**From Corpora.** Pseudo-labels derived from datasets are generally domain-specific, limiting their applicability for cross-domain training. Cross-domain keyphrase generation, aiming to generalize models across diverse domains, remains a significant challenge. Where data is limited, external corpora offer a viable solution for generating pseudo-labels.

To address cross-domain keyphrase generation, the OpenDomainKP model [12] extracts grammatically valid noun phrases and their contexts from external corpora, constructing a phrase repository. For each phrase $z$, its context is encoded as $v_z$. Similarly, the input text $x$ is encoded as $v_x$, and the cosine similarity between $v_z$ and $v_x$ is computed as their relevance. The top-k most relevant phrases are then retrieved from the repository, forming a pseudo-label set that incorporates external knowledge for model training. The OpenDomainKP model's key strength lies in its ability to perform cross-domain unsupervised keyphrase generation with minimal modifications. By simply expanding the phrase repository, it adapts to diverse domains without requiring domain-specific training data. This flexibility opens new avenues for advancing keyphrase generation, motivating further exploration of effective strategies to leverage external corpora.

### 4.2 Large Language Model-based Unsupervised Keyphrase Generation

Keyphrase generation requires summarizing a source document using vocabulary that may not explicitly appear in the text, necessitating a profound understanding of domain-specific knowledge. Previous unsupervised approaches have attempted to address this challenge by using domain-specific pretraining. For example, the SciBART [50], which was pretrained from scratch on a large-scale scientific dataset. However, domain-specific pretraining is costly, and the obtained models are not easily transferable across different domains.

In contrast, large language models (LLMs) have richer pretraining corpora and more diverse pretraining tasks. They have outperformed traditional PLMs on zero-shot tasks [36], bringing new opportunities for keyphrase generation. Recent research has primarily focused on leveraging prompt learning to generate keyphrases in a zero-shot manner.

**Prompt Learning.** Leveraging multi-domain and multi-task training, large language models generate keyphrases in an unsupervised manner using prompts, delivering superior performance in semantic-based evaluations compared to state-of-the-art unsupervised models, and achieving results comparable to those of supervised models. Researchers [41] explore the zero-shot keyphrase generation capabilities of ChatGPT, focusing on prompt template design and evaluating the diversity of generated results.

Due to the constraints of input length, most models generate keyphrases based on truncated text rather than the full document. In contrast, ChatGPT allows longer input tokens, and experiments on long document datasets show that ChatGPT outperforms all baseline models in keyphrase generation for long documents [29].

Instead of just depending on a single-step prompt, Researchers [48] attempt to employ LLMs more comprehensively. They simulate the process by which an

article author selects keyphrases, prompting LLM to 1) directly generate candidate phrases from the document; 2) expand candidate phrases into their hypernyms or synonyms; 3) retrieve related candidate phrases from similar documents; 4) rank all candidate phrases, forming the final keyphrases. This method leverages the zero-shot generation capability, extended token limits, and rich domain knowledge of LLMs, providing a new perspective for keyphrase generation.

LLMs fine-tuned with instructions can adapt well to human commands and perform a variety of natural language processing tasks. However, when generating keyphrases, LLMs often mistakenly lean toward performing "named entity recognition" tasks, extracting all entities present in the source document. To correct this, researchers [51] designed a novel self-consistency decoding process, utilizing frequency information to capture the phrases that convey the most important information, improving generation results when tested on GPT-3.5-turbo and GPT-4. It is important to address the "hallucination" problem in LLMs to enhance the accuracy of keyphrase generation.

Although LLMs demonstrate strong performance in generating keyphrases from zero-shot prompts, identifying the optimal prompt remains a challenging task. Additionally, processing domain-specific texts presents ongoing difficulties. Techniques such as prompt engineering and retrieval-augmented generation can be employed to enhance performance.

## 5 Evaluation Metrics and Datasets

### 5.1 Evaluation Metrics

Keyphrase prediction models can be evaluated in three ways: reference-based metrics, reference-free metrics, and human evaluation. Reference-based metrics compare the predicted keyphrase set with a reference set (label data), requiring high-quality labels. Reference-free metrics do not rely on labels but need to be task-specifically designed. Human evaluation reflects human judgment but requires careful design to ensure consistency and reproducibility. Below, we describe these three types of evaluation metrics.

**Reference-based Evaluation.** A high-quality reference set can act as a representative summary of the key information in the source document. Reference-based evaluation methods assess model performance by measuring the alignment between the predicted sets and label sets, which can be calculated by lexical or semantic matching.

**1) Lexical Matching.** Common metrics include Precision, Recall, and F1-score, which quantify the phrase-level lexical matching (Eq. 2). Here, $TP$ refers to correctly identified keyphrases, $FP$ denotes non-keyphrases incorrectly predicted as keyphrases, and $FN$ represents keyphrases missed by the model. In practice, $Recall@k$ and $F1@k$ are commonly used, where $k$ is the number of keyphrases for evaluation. Table 2 shows the $F1@5$ and $F1@10$ performance of some recent representative works on Inspec [18] and SemEval-2010 [22] datasets.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad \text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{2}$$

**Table 2.** Performance on Inspec and SemEval-2010 datasets.

| | Methods | Inspec | | SemEval-2010 | |
|---|---|---|---|---|---|
| | | F1@5 | F1@10 | F1@5 | F1@10 |
| **Present Keyphrases** | **YAKE** | 18.08 | 19.62 | 11.76 | 14.40 |
| | **JointGL** | 32.61 | 40.17 | 13.02 | 19.35 |
| | **MDERank** | 27.85 | 34.36 | 13.05 | 18.27 |
| | **HyperRank** | 33.35 | 40.79 | 14.79 | 21.33 |
| | **SAMRank** | 33.96 | 39.35 | 15.28 | 18.36 |
| | **PromptRank** | 31.73 | 37.88 | 17.24 | 20.66 |
| | **AutoKeyGen** | 30.30 | 34.50 | 18.70 | 24.00 |
| | **KeyBART** | 30.72 | - | 20.25 | - |
| | **ChatGPT** | 40.10 | - | 26.20 | - |
| **Absent Keyphrases** | **AutoKeyGen** | - | 1.70(R@10) | - | 1.00(R@10) |
| | **KeyBART** | 1.83 | - | 1.12 | - |
| | **ChatGPT** | 2.90 | - | 0.50 | - |

*Note: Baseline model results are taken from original papers or other studies [23,25,30,43], and ChatGPT's results are from [41].

Based on the fundamental metrics, researchers [54] noted the value of $k$ in $F1@k$ is typically fixed at 5, 10,or 15. However, the number of keyphrases in both the predicted set and the label set is not fixed. Therefore, they proposed the $F1@O$ and $F1@M$ metrics, where $O$ represents the actual size of the label set, and $M$ is the actual size of the predicted set. Although widely used, the F1-score has limitations in capturing deep semantic meaning, making it a relatively weak measurement metric. Consequently, semantic matching has been explored as an alternative.

**2) Semantic Matching.** SoftKeyScores [26] is an evaluation system that includes the Keyphrase Match Rate (KMR) and the Score. The lexical metric KMR is an adaptation of the Translation Edit Rate (TER), which measures the edit distance between two sets of phrases. The semantic metric Score is similar to BERTScore [56], using embeddings $g_i$ and $l_j$ from the predicted and label embedding sets $G$ and $L$, applying greedy matching to calculate the maximum similarity between phrases. The final score, $F_{score}$, is the harmonic mean of $P_{score}$ and $R_{score}$ (Eq. 3).

$$P_{\text{score}} = \frac{1}{|G|} \cdot \sum_{g_i \in G} \max_{l_j \in L} \text{score}\left(g_i, l_j\right), \quad R_{\text{score}} = \frac{1}{|L|} \cdot \sum_{l_j \in L} \max_{g_i \in G} \text{score}\left(g_i, l_j\right) \tag{3}$$

KPEval [52] includes four metrics (reference agreement, faithfulness, diversity, and utility). Reference agreement is a label-based semantic evaluation that shares a similar design idea with SoftKeyScores [26] and employs cosine similarity to measure the relationship between two phrase embeddings.

**Reference-free Evaluation.** Labels in datasets are typically annotated by authors or experts. Authors with a deeper understanding of the domain tend to provide absent keyphrases that summarize the document. In contrast, experts, whose domain knowledge may vary, tend to focus on more detailed aspects of the document. As a result, the labels can be highly subjective. Evaluating keyphrase prediction results solely based on the label sets may be unreliable. To address this, reference-free evaluation metrics have been adopted to offer more effective and robust assessments.

**1) Existing Metrics.** As mentioned above, the KPEval [52] includes three reference-free metrics. Firstly, the Faithfulness reflects whether keyphrases are semantically grounded in source documents. Absent keyphrases are considered faithful if they are synonyms, hypernyms, or hyponyms of concepts in the document. Present keyphrases are considered faithful if their boundaries are accurately identified (e.g., correctly extracting "NP-hard problem" rather than just "hard problem"). Secondly, the Diversity refers whether predicted keyphrases includes diverse keyphrases with minimal repetitions, calculated based on lexical repetition percentage and semantic similarity between phrases. It is calculated based on two components: lexical repetition percentage and average semantic similarity between phrases. Thirdly, the Utility is measured by the impact of the keyphrase set on downstream information retrieval performance.

**2) A New Metric.** Based on the analysis of keyphrase properties in Sect. 2, we propose an unsupervised keyphrase evaluation metric, termed ***Coverage***, which assesses whether a keyphrase set represents the topics of the source document without omitting critical information. Specifically, pretrained language models are employed to encode the source document at three levels(title, sentence, and full text), yielding a set of vector representations $X(x_i \in X)$. Concurrently, the generated keyphrase set is encoded to corresponding embedding set $G(g_i \in G)$. In contrast to the "Score" metric in SoftKeyScores, the reference set for the predicted keyphrases is not the reference set but the multi-level semantic embeddings derived from the source document. The final metric $F_{Coverage}$ is calculated as the harmonic mean of $P_{Coverage}$ and $R_{Coverage}$ (Eq. 4).

$$P_{\text{Coverage}} = \frac{1}{|X|} \sum_{g_i \in G} \max_{x_j \in X} \frac{x_j^\top \hat{g}_i}{\|x_j\| \, \|\hat{g}_i\|} \quad R_{\text{Coverage}} = \frac{1}{|G|} \sum_{x_j \in X} \max_{g_i \in G} \frac{x_j^\top \hat{g}_i}{\|x_j\| \, \|\hat{g}_i\|} \tag{4}$$

**Human Evaluation.** Due to the limitations of automatic metrics, human evaluation is often used to assess the quality of predicted keyphrases. For a long time, human evaluation has been regarded as the golden standard for the quality of experimental results, with its authenticity rarely questioned. However, a recent study [2] reported a review of human evaluation experiments in NLP papers over the past five years, finding that their reproducibility was around 5%. Even when the original authors were willing to provide assistance, the reproducibility of human evaluation experiments was only 20%. Given the absence of a universal evaluation guideline for human evaluation on keyphrase prediction, ensuring

**Table 3.** Statistics of datasets

| Dataset | Domain | Counts | Length | Annotator | Year |
|---|---|---|---|---|---|
| **Short-text Dataset( length<=500 )** | | | | | |
| **INSPEC** [18] | Comp.Science | 2000 | 128 | E | 2003 |
| **SemEval-2017** [1] | Science | 500 | 178 | E | 2017 |
| **KP20k** [30] | Comp.Science | 568k | 176 | A | 2017 |
| **STACKEX** [54] | Comp.Science | 331k | 300 | A | 2019 |
| **KP-Biomed** [17] | Biomedical | 5.9M | 271 | A | 2022 |
| **Long-text Dataset( length>500 )** | | | | | |
| **NUS** [33] | Science | 211 | 7644 | A&R | 2007 |
| **DUC2001** [47] | News | 308 | 740 | R | 2008 |
| **Krapivin2009** [24] | Comp.Science | 2304 | 8040 | A | 2009 |
| **SemEval-2010** [22] | Multi-domain | 244 | 7961 | A&R | 2010 |
| **OpenKP** [53] | Multi-domain | 148k | 900 | E | 2019 |
| **KPTimes** [15] | News | 280k | 921 | E | 2019 |
| **LDKP3K** [28] | Science | 100K | 6027 | A | 2021 |
| **LDKP10K** [28] | Science | 1.3M | 4384 | A | 2021 |
| **METAKP** [51] | Multi-domain | 7500 | Mixed | GPT-4&R | 2024 |
| **Multi-modal/Multi-lingual Dataset** | | | | | |
| **Tweet-KP** [49] | Multi-modal | 53781 | 27 | A | 2020 |
| **Papyrus** [35] | Multi-lingual | 16427 | 290–573 | A | 2022 |
| **EUROPA** [37] | Multi-lingual | 285k | 5220 | A | 2024 |

*Note: A for authors; R for Readers; E for Experts

high consistency between evaluators and acceptable reproducibility remains a significant challenge.

### 5.2 Datasets

The datasets for the keyphrase prediction task cover various domains such as computer science, news, and biomedicine, with annotations made by authors, readers, or experts. We also investigate emerging cross-lingual and multimodal keyphrase prediction datasets. Table 3 provides detailed statistical information about these datasets, categorized by short-text datasets, long-text datasets, and multi-modal/multi-lingual datasets, arranged chronologically.

## 6 Challenges and Future Directions

### 6.1 Absent Keyphrase Generation

Generating absent keyphrases is a challenging task that requires a deep understanding of semantics and domain-specific knowledge. Absent keyphrases were initially defined as phrases that do not match any contiguous subsequence in the document [30]. Then, according to [6], the phrases have been reclassified into three types: 1) Reordered keyphrases (constituent words in the text but not contiguous); 2) Mixed keyphrases (some constituent words in the text); 3) Unseen

keyphrases (no constituent words in the text). Current models primarily generate reordered keyphrases and place less emphasis on mixed or unseen keyphrases, which should be prioritized in future research.

Large language models can generate high-quality absent keyphrases that perform well in both automatic metrics and human preferences. However, challenges still remain. For instance, the generated keyphrases are basically a rewrite of the phrases in the source document. The hallucination problem, where the model generates incorrect or irrelevant phrases, further undermines result accuracy. Enhancing keyphrase accuracy in LLMs through prompt engineering, in-context learning, and retrieval-augmented generation is a promising research direction.

### 6.2 Keyphrase Evaluation

Current widely used automatic evaluation metrics, such as F1-score and Recall, were not specifically designed for the keyphrase prediction task and rely heavily on high-quality annotations, limiting their effectiveness. Although many studies have sought to improve these methods [26,52,54], they remain largely dependent on label-based evaluation.

Keyphrases should be noun phrases that reflect the content of the source document, preserving its theme and essential information without altering boundary words. Based on the analysis in Sect. 2, keyphrase prediction can be evaluated in an unsupervised manner from three dimensions: conciseness, coverage, and distinctiveness. Conciseness and distinctiveness apply to individual keyphrases, while coverage pertains to the entire keyphrase set. Our future work will focus on evaluating keyphrases from these three perspectives.

## 7 Conclusion

Unsupervised keyphrase prediction has gained significant improvements with the advent of pretrained language models. To comprehensively summarize recent advancements in these methods, this survey provides an in-depth analysis of the entire task pipeline. We examine the intrinsic characteristics of keyphrases from a linguistic perspective, offering valuable insights for model design and subsequent evaluation. Furthermore, we present a detailed discussion of unsupervised keyphrase extraction and generation methods, encompassing approaches ranging from statistical techniques to deep learning, with an emphasis on emerging advancements. Additionally, we systematically review evaluation metrics for keyphrase prediction, guiding researchers in selecting appropriate metrics or designing new ones. Lastly, we share our perspectives on future research of keyphrase prediction, aiming to inspire further progress in this field.

**Acknowledgements.** This work was supported by the Innovative Development Joint Fund Key Projects of Shandong NSF (ZR2022LZH007), the National Natural Science Foundation of China (62376138).

## References


1. Augenstein, I., Das, M., Riedel, S., Vikraman, L., McCallum, A.: Semeval 2017 task 10: scienceie-extracting keyphrases and relations from scientific publications. arXiv preprint arXiv:1704.02853 (2017)
2. Belz, A., Thomson, C., Reiter, E., Mille, S.: Non-repeatable experiments and non-reproducible results: the reproducibility crisis in human evaluation in nlp. In: Findings of the Association for Computational Linguistics: ACL 2023, pp. 3676–3687 (2023)
3. Bennani-Smires, K., Musat, C., Hossmann, A., Baeriswyl, M., Jaggi, M.: Simple unsupervised keyphrase extraction using sentence embeddings. arXiv preprint arXiv:1801.04470 (2018)
4. Boudin, F.: Unsupervised keyphrase extraction with multipartite graphs. arXiv preprint arXiv:1803.08721 (2018)
5. Boudin, F., Aizawa, A.: Unsupervised domain adaptation for keyphrase generation using citation contexts. arXiv preprint arXiv:2409.13266 (2024)
6. Boudin, F., Gallina, Y.: Redefining absent keyphrases and their effect on retrieval effectiveness. arXiv preprint arXiv:2103.12440 (2021)
7. Boudin, F., Gallina, Y., Aizawa, A.: Keyphrase generation for scientific document retrieval. arXiv preprint arXiv:2106.14726 (2021)
8. Bougouin, A., Boudin, F., Daille, B.: Topicrank: graph-based topic ranking for keyphrase extraction. In: International Joint Conference on Natural Language Processing (2013)
9. Campos, R., Mangaravite, V., Pasquali, A., Jorge, A., Nunes, C., Jatowt, A.: Yake! keyword extraction from single documents using multiple local features. Inf. Sci. **509**, 257–289 (2020)
10. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: pre-training of deep bidirectional transformers for language understanding. ArXiv **abs/1810.04805** (2019)
11. Ding, H., Luo, X.: Attentionrank: unsupervised keyphrase extraction using self and cross attentions. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pp. 1919–1928 (2021)
12. Do, L.T., Akash, P.S., Chang, K.C.C.: Unsupervised open-domain keyphrase generation. arXiv preprint arXiv:2306.10755 (2023)
13. Firoozeh, N., Nazarenko, A., Alizon, F., Daille, B.: Keyword extraction: issues and methods. Nat. Lang. Eng. **26**(3), 259–291 (2020)
14. Florescu, C., Caragea, C.: Positionrank: an unsupervised approach to keyphrase extraction from scholarly documents. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (volume 1: long papers), pp. 1105–1115 (2017)
15. Gallina, Y., Boudin, F., Daille, B.: Kptimes: a large-scale dataset for keyphrase generation on news documents. arXiv preprint arXiv:1911.12559 (2019)
16. Gollapalli, S.D., Caragea, C.: Extracting keyphrases from research papers using citation networks. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 28 (2014)
17. Houbre, M., Boudin, F., Daille, B.: A large-scale dataset for biomedical keyphrase generation. arXiv preprint arXiv:2211.12124 (2022)
18. Hulth, A.: Improved automatic keyword extraction given more linguistic knowledge. In: Conference on Empirical Methods in Natural Language Processing (2003)
19. Joshi, R., Balachandran, V., Saldanha, E., Glenski, M., Volkova, S., Tsvetkov, Y.: Unsupervised keyphrase extraction via interpretable neural networks. arXiv preprint arXiv:2203.07640 (2022)

20. Kang, B., Shin, Y.: Samrank: unsupervised keyphrase extraction using self-attention map in bert and gpt-2. In: The 2023 Conference on Empirical Methods in Natural Language Processing (2023)
21. Kang, B., Shin, Y.: Improving low-resource keyphrase generation through unsupervised title phrase generation. In: Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), pp. 8853–8865 (2024)
22. Kim, S.N., Medelyan, O., Kan, M.Y., Baldwin, T.: Semeval-2010 task 5 : automatic keyphrase extraction from scientific articles. In: *SEMEVAL (2010)
23. Kong, A., et al.: Promptrank: unsupervised keyphrase extraction using prompt. arXiv preprint arXiv:2305.04490 (2023)
24. Krapivin, M., Autaeu, A., Marchese, M., et al.: Large dataset for keyphrases extraction (2009)
25. Kulkarni, M., Mahata, D., Arora, R., Bhowmik, R.: Learning rich representation of keyphrases from text. arXiv preprint arXiv:2112.08547 (2021)
26. Kundu, T., Chowdhury, J.R., Caragea, C.: Neural keyphrase generation: analysis and evaluation. arXiv preprint arXiv:2304.13883 (2023)
27. Liang, X., Wu, S., Li, M., Li, Z.: Unsupervised keyphrase extraction by jointly modeling local and global context. arXiv preprint arXiv:2109.07293 (2021)
28. Mahata, D., et al.: Ldkp: a dataset for identifying keyphrases from long scientific documents. arXiv preprint arXiv:2203.15349 (2022)
29. Martínez-Cruz, R., López-López, A.J., Portela, J.: Chatgpt vs state-of-the-art models: a benchmarking study in keyphrase generation task. arXiv preprint arXiv:2304.14177 (2023)
30. Meng, R., Zhao, S., Han, S., He, D., Brusilovsky, P., Chi, Y.: Deep keyphrase generation. arXiv preprint arXiv:1704.06879 (2017)
31. Merrouni, Z.A., Frikh, B., Ouhbi, B.: Hake: an unsupervised approach to automatic keyphrase extraction for multiple domains. Cogn. Comput. **14**(2), 852–874 (2022)
32. Nguyen, T.D., Kan, M.Y.: Keyphrase extraction in scientific publications. In: International conference on Asian digital libraries, pp. 317–326. Springer (2007)
33. Nguyen, T.D., Kan, M.Y.: Keyphrase extraction in scientific publications. In: International Conference on Asian Digital Libraries (2007)
34. Ouyang, L., et al.: Training language models to follow instructions with human feedback. Adv. Neural. Inf. Process. Syst. **35**, 27730–27744 (2022)
35. Piedboeuf, F., Langlais, P.: A new dataset for multilingual keyphrase generation. Adv. Neural. Inf. Process. Syst. **35**, 38046–38059 (2022)
36. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al.: Language models are unsupervised multitask learners. OpenAI blog **1**(8), 9 (2019)
37. Salaün, O., Piedboeuf, F., Berre, G.L., Hermelo, D.A., Langlais, P.: Europa: a legal multilingual keyphrase generation dataset. arXiv preprint arXiv:2403.00252 (2024)
38. Shen, X., Wang, Y., Meng, R., Shang, J.: Unsupervised deep keyphrase generation. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, pp. 11303–11311 (2022)
39. Song, M., Feng, Y., Jing, L.: A preliminary empirical study on prompt-based unsupervised keyphrase extraction. arXiv preprint arXiv:2405.16571 (2024)
40. Song, M., Geng, X., Yao, S., Lu, S., Feng, Y., Jing, L.: Large language models as zero-shot keyphrase extractor: a preliminary empirical study. arXiv preprint arXiv:2312.15156 (2023)
41. Song, M., et al.: Is chatgpt a good keyphrase generator? a preliminary study. arXiv preprint arXiv:2303.13001 (2023)

42. Song, M., Liu, H., Feng, Y., Jing, L.: Improving embedding-based unsupervised keyphrase extraction by incorporating structural information. In: Findings of the Association for Computational Linguistics: ACL 2023, pp. 1041–1048 (2023)
43. Song, M., Liu, H., Jing, L.: Hyperrank: hyperbolic ranking model for unsupervised keyphrase extraction. In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pp. 16070–16080 (2023)
44. Song, M., Xu, P., Feng, Y., Liu, H., Jing, L.: Mitigating over-generation for unsupervised keyphrase extraction with heterogeneous centrality detection. In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pp. 16349–16359 (2023)
45. Sparck Jones, K.: A statistical interpretation of term specificity and its application in retrieval. J. Documentation **28**(1), 11–21 (1972)
46. Sun, Y., Qiu, H., Zheng, Y., Wang, Z., Zhang, C.: Sifrank: a new baseline for unsupervised keyphrase extraction based on pre-trained language model. IEEE Access **8**, 10896–10906 (2020)
47. Wan, X., Xiao, J.: Single document keyphrase extraction using neighborhood knowledge. In: AAAI Conference on Artificial Intelligence (2008)
48. Wang, S., Dai, S., Jiang, J.: Thinking like an author: a zero-shot learning approach to keyphrase generation with large language model. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 335–350. Springer (2024)
49. Wang, Y., Li, J., Lyu, M.R., King, I.: Cross-media keyphrase prediction: a unified framework with multi-modality multi-head attention and image wordings. arXiv preprint arXiv:2011.01565 (2020)
50. Wu, D., Ahmad, W.U., Chang, K.W.: Rethinking model selection and decoding for keyphrase generation with pre-trained sequence-to-sequence models. arXiv preprint arXiv:2310.06374 (2023)
51. Wu, D., Shen, X., Chang, K.W.: Metakp: on-demand keyphrase generation. arXiv preprint arXiv:2407.00191 (2024)
52. Wu, D., Yin, D., Chang, K.W.: Kpeval: towards fine-grained semantic-based keyphrase evaluation. arXiv preprint arXiv:2303.15422 (2023)
53. Xiong, L., Hu, C., Xiong, C., Campos, D., Overwijk, A.: Open domain web keyphrase extraction beyond language modeling. arXiv preprint arXiv:1911.02671 (2019)
54. Yuan, X., et al.: One size does not fit all: generating and evaluating variable number of keyphrases. arXiv preprint arXiv:1810.05241 (2018)
55. Zhang, L., et al.: Mderank: a masked document embedding rank approach for unsupervised keyphrase extraction. arXiv preprint arXiv:2110.06651 (2021)
56. Zhang, T., Kishore, V., Wu, F., Weinberger, K.Q., Artzi, Y.: Bertscore: evaluating text generation with bert. arXiv preprint arXiv:1904.09675 (2019)
57. Zhang, Z., Liang, X., Zuo, Y., Lin, C.: Improving unsupervised keyphrase extraction by modeling hierarchical multi-granularity features. Inf. Process. Manage. **60**(4), 103356 (2023)