A Survey of Keyphrase Generation

Anonymous ACL submission

Abstract

Keyphrase generation refers to the task of producing a set of words or phrases that summarises the content of a document. Continuous efforts have been dedicated to this task over the past few years, spreading across multiple lines of research, such as model architectures, data resources, and use-case scenarios. Yet, the current state of keyphrase generation remains unknown as there has been no attempt to review and analyse previous work. This survey bridges that gap and provides a comprehensive overview of the recent progress, limitations and open challenges in keyphrase generation. Our analysis of over 40 research papers reveals interesting new insights, such as that 1) commonly-used datasets are so similar that there is no practical benefit in using them together for evaluation, or that 2) the performance of many models was significantly overestimated due to the application of normalization procedures in ground truth. This paper not only surveys the literature but also addresses some of these concerns by training, documenting and releasing a strong PLM-based model for keyphrase generation, along with an evaluation framework, as an effort to facilitate future research.

1 Introduction

Keyphrase generation involves generating a set of words or phrases that summarise the content of a source document. These so-called keyphrases concisely and explicitly encapsulate the core content of a document, which makes them valuable for a variety of NLP and information retrieval tasks. For instance, keyphrases were proven useful for improving document indexing (Fagan, 1987; Zhai, 1997; Jones and Staveley, 1999; Gutwin et al., 1999; Boudin et al., 2020), summarization (Zha, 2002; Wan et al., 2007; Liu et al., 2021; Koto et al., 2022) and question-answering (Subramanian et al., 2018; Yang et al., 2019; Lee et al., 2021), analyzing topic evolution (Hu et al., 2019; Cheng et al., 2020; Lu et al., 2021) or assisting with reading comprehension (Chi et al., 2007; Jiang et al., 2023).

The task of keyphrase generation was initially introduced by Liu et al. (2011) as an extension of keyphrase extraction, which involves identifying the most important phrases within a document. The added value of keyphrase generation lies in its ability to produce *keyphrases that are absent from the source document*. This ability is particularly important when the source document is short and may lack appropriate keyphrases. This motivated the canonical work of Meng et al. (2017), which introduced a sequence-to-sequence learning approach to keyphrase generation. Their proposed model, named CopyRNN, builds upon an RNN encoder-decoder architecture (Cho et al., 2014; Sutskever et al., 2014) and incorporates a copying mechanism (Gu et al., 2016) that enables the model to identify important phrases within the source text. Perhaps more importantly, they introduced the KP20k dataset which contains more than 500K keyphrase-annotated samples and allows the training of neural models in an end-to-end manner.

Over the past few years, continuous efforts have been devoted to improve the effectiveness of keyphrase generation models. These efforts have been spread across different lines of research, such as model architectures, data resources, and use-case scenarios, often pursued separately. This survey presents an overview of the current state of keyphrase generation, discussing recent progress, remaining limitations and open challenges. More specifically, we compiled and analysed a collection of over 40 papers on keyphrase generation, identifying the type(s) of contribution these papers made (§3), examining the most frequently used benchmark datasets (§3.1) and evaluation metrics (§3.2), providing descriptions of proposed models while highlighting important milestones (§3.3), and investigating how proposed models perform against each other (§3.4).

Our analysis reveals that: 1) there is a gap in the literature regarding papers focusing on data analysis and reproduction studies, 2) reporting results on several commonly used datasets offers no practical benefit compared to using only KP20k, 3) the performance of many models may be overestimated due to discrepancies in the normalization of ground truth keyphrases, 4) dedicated models have been superseded by fine-tuned pre-trained language models (PLMs), yet the overall performance gain since early models remains limited, and 5) the limited availability of pre-trained models not only impedes progress but also obstructs reproducibility and fair comparison with previous work.

Our work goes beyond surveying the existing literature and addresses some of the aforementioned concerns by training, documenting and releasing a strong PLM-based model for keyphrase generation along with an evaluation framework to facilitate future research (§4). Finally, we discuss some of the open challenges in keyphrase generation and propose actionable directions to address them (§5).

2 Survey Scope

Our survey encompasses a total of 44 research papers selected based on the following criteria: they are accessible through the ACL Anthology, they contain the phrase "*keyphrase generation*" either in their titles or abstracts, and they have been published after the canonical work of Meng et al. (2017). For a more comprehensive coverage, we also include papers from other NLP-related venues, comprising AAAI (4 papers), SIGIR (1 paper), and CIKM (1 paper). To keep the number of papers manageable, we arbitrarily disregard papers from pre-print servers (e.g. arXiv) or those published in non-ACL journals (e.g. Natural Language Engineering). Nonetheless, we are confident that our sample represents a comprehensive portion of the research on keyphrase generation, encompassing all papers published at major NLP venues in the last seven years. This includes, for instance, the ten most cited articles in the field.[1]

For each paper in our sample, we manually collect the following information:

- The **type(s) of contribution** the paper is making. We adopt the ACL 2023 classification of contribution types (Rogers et al., 2023), which includes: 1) NLP engineering experiment (most papers proposing methods to improve state-of-the-art), 2) approaches for low–compute settings, efficiency, 3) approaches for low-resource settings, 4) data resources, 5) data analysis, 6) model analysis and interpretability, 7) reproduction studies, 8) position papers, 9) surveys, 10) theory, 11) publicly available software and pre-trained models.
- For papers proposing models, we record their **best scores** on each dataset they experiment with, in the form of $\langle dataset, metric, value\rangle$ triples. We extract scores primarily from the main tables of the content, supplementing with tables from appendices only if they report superior performance. In cases where multiple model variants are reported, we select the one demonstrating the best overall performance, or, when it is not clear, the one that performs best on the KP20k dataset. In total, we extracted 700 triples from our sample, corresponding to 42 distinct models.
- We also document the **architecture** of the proposed models (e.g RNNs, Transformers), the use of **statistical significance tests** on the results, and the availability of both the **code** and the **model weights**.

All the data collected in the course of this study is available at `www.github.com/anonymous`.

Related Surveys

To our knowledge, this is the first attempt at compiling and analyzing the performance of keyphrase generation models. In contrast, several surveys have been carried out on keyphrase extraction, starting with (Hasan and Ng, 2014), which focused on pre-deep-learning unsupervised methods. Subsequent surveys, such as (Çano and Bojar, 2019), (Papagiannopoulou and Tsoumakas, 2020) and (Firoozeh et al., 2020), included additional, more recent methods and presented comparative experimental studies. More recently, Song et al. (2023) carried out a comprehensive review of keyphrase extraction methods, covering PLM-based models, and Xie et al. (2023) performed a large-scale analysis of keyphrase prediction methods, which included results from some generative models. Despite marked differences, notably in the model architectures and training procedures, previous research on keyphrase extraction and generation converge on the datasets and evaluation metrics, making these surveys complementary to ours.

[1] `https://www.semanticscholar.org/search?q="keyphrase%20generation"&sort=total-citations`

3 Analysis

We start our analysis by presenting statistics on the types of contribution made in the papers we considered for our survey (see Table 1). Most of the papers propose new models for keyphrase generation (86.4%), suggesting that the primary emphasis within the field is on improving the performance of the state-of-the-art. This is reinforced by the fact that the second most common contribution is data resources (18.2%), essential for validating improvements. Additionally, some attention was given to model analysis and interpretability (13.6%), particularly through empirical evaluations of multiple models (Çano and Bojar, 2019; Meng et al., 2021, 2023; Wu et al., 2023) and evaluations via downstream tasks (Boudin et al., 2020; Boudin and Gallina, 2021). Our analysis also underscores a *gap in the literature regarding papers that concentrate on data analysis, reproduction studies and surveys*. This survey paper bridges this gap by, among other aspects, offering a fresh perspective on the complementarity of existing datasets, conducting replication experiments on model evaluation, and thoroughly documenting the training process of a strong baseline model for keyphrase generation.

Type of contribution	%
NLP engineering experiment	86.4
Data resources	18.2
Model analysis and interpretability	13.6
Software and pre-trained models	9.1
Approaches for low-resource settings	9.1
Approaches for low-compute settings	2.3

Table 1: Percentage of papers (%) in our sample that make each type of contribution. A paper can make one or more types of contributions.

3.1 Benchmark Datasets

We proceed with our analysis by examining the most frequently used datasets (see Figure 1, detailed statistics of the datasets are provided in §A.1). We find that 23 distinct datasets were employed across the examined papers, with five datasets notably more prevalent than others: KP20k (Meng et al., 2017), SemEval-2010 (Kim et al., 2010), Inspec (Hulth, 2003), Krapivin (Krapivin et al., 2009), and NUS (Nguyen and Kan, 2007). These datasets are commonly used together in papers, with 19 out of 42 papers (45.2%) employing all five, and 33 out of 42 (78.6%) employing at least two of them. The five datasets exclusively contain scientific abstracts, while the remaining datasets encompass various sources such as news, social media and web pages. This strong domain bias can be attributed to two main factors: the ready availability of scientific abstracts, and the frequent presence of author-assigned keyphrases associated with them, serving as naturally occurring ground truth. When considering size, only a handful of datasets contain a sufficient number of samples (i.e. $> 100k$ training samples, underlined in Figure 1) to effectively train generative models. Thus, the majority of these datasets are relatively small (i.e. $< 1k$ samples) and used for testing purposes only.

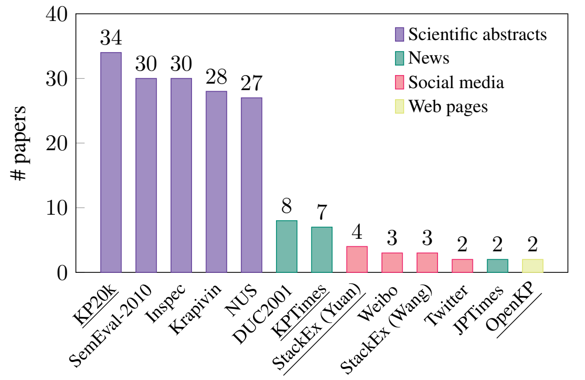

Figure 1: Number of papers utilizing each dataset. Underlined datasets contain $100K$+ training samples. Datasets used only once are omitted for clarity.

A closer examination of the five widely-used datasets reveals substantial similarities among them. For instance, they all contain scientific abstracts from the Computer Science domain, and at least three of them –KP20k, SemEval-2010, and Krapivin– include documents from the same source (ACM Digital Library). Conversely, they differ notably in their ground truth: two contain author-assigned keyphrases (KP20k and Krapivin), two feature a combination of author- and reader-assigned keyphrases (SemEval-2010 and NUS), and the last includes indexer-assigned keyphrases (Inspec). This raises *questions about the practicality of using them together in experiments, as well as the potential for data leakage between them*. To shed light on these questions, we measured the correlation between the model scores across datasets, exploring whether models perform uniformly across different datasets. Our objective here is to determine the extent to which including more than one of these datasets in the experiments of a paper provides additional insights. From the correlation matrix in Figure 2, we see that the performance of models among the five widely-used datasets is almost perfectly correlated (Pearson's correlation coefficient $\rho > 0.9$, p-value < 0.01). This observation implies that *there is no practical benefit in reporting the results on more than one of these five datasets*, despite the common practice among previous studies of doing so. Therefore, our findings advocate that conducting experiments only on KP20k is sufficient, considering its broader adoption in previous work and its larger size compared to other datasets.

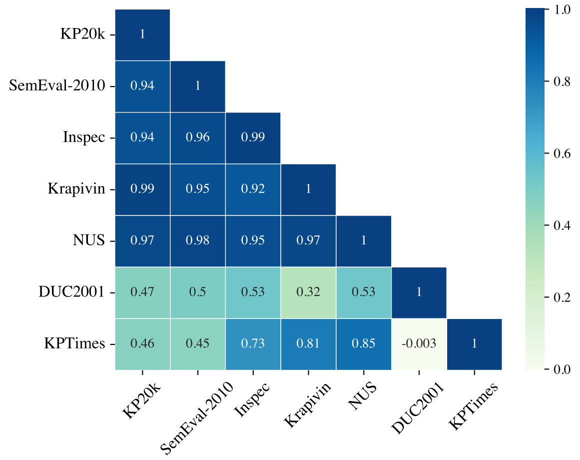

Figure 2: Pearson's correlation coefficient ρ computed between the model scores across datasets.

3.2 Evaluation Metrics

We move forward with our analysis by examining the evaluation of automatically generated keyphrases within our sample of papers. With the exception of (Wu et al., 2022b), all the proposed models are solely assessed through intrinsic evaluation, which involves comparing their output with a single ground truth using exact matching. From the extracted score triples, we find that 40 distinct evaluation metrics were reported across the papers we examined (see Figure 3, detailed definitions of the evaluation metrics are provided in §A.2). The majority of papers describing models (33 out of 42, 78.6%) provide separate results for present and absent keyphrases, following the methodology of (Meng et al., 2017). As for the metrics, there is a high degree of consensus on the F_1 measure for present keyphrases, with two configurations standing out: $F_1@M$ (using all the keyphrases predicted by the model) and $F_1@k$ (using the top-k predicted keyphrases, with $k \in \{5, 10\}$). However, the situation is less clear for absent keyphrases, which are more challenging to predict and therefore result in very low scores, with the F_1 measure being used alongside with the recall at a large number of predicted keyphrases ($k \in \{10, 50\}$).

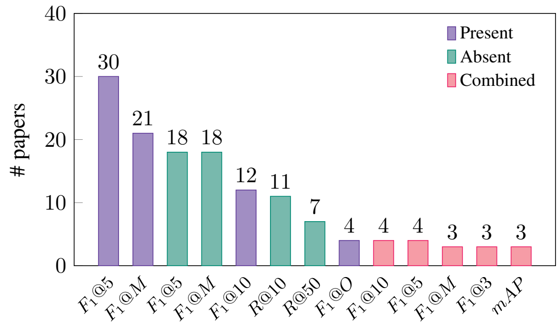

Figure 3: Number of papers employing each evaluation metric. Metrics used < 3 times are excluded for clarity.

Upon closer inspection of the evaluation settings in our sample of papers, we find that some form of normalization procedure is frequently applied prior to computing evaluation metrics, as observed in at least 22 out of 42 papers (52.4%). This procedure, commonly referred to as Meng et al. (2017)'s preprocessing[2], is applied to ground-truth keyphrases and involves the following steps: 1) removing all the abbreviations/acronyms in parentheses, 2) tokenizing on non-letter characters, and 3) replacing digits with symbol `<digit>`. This normalization impacts the evaluation (see an example in Table 3 in §A.3), *potentially leading to an overestimation of model performance and jeopardizing comparability with studies that do not employ it*. To gain insights on this issue, we conducted a series of replication experiments by reassessing the performance of three models –catSeqTG-2RF1 (Chan et al., 2019), ExHiRD-h (Chen et al., 2020) and SetTrans (Ye et al., 2021b)– for which the authors stated that they applied this normalization and provided the outputs of their model.

From the results in Figure 4, we observe that applying the normalization procedure significantly increases the scores for the majority of the evaluation metrics. The impact of the normalization procedure is more pronounced for present keyphrases, showing an absolute difference of +2.2 points ($F_1@M$) and +3.5 points ($F_1@5$). We notice a some difference in scores between the original ( ) and our

[2] https://github.com/memray/OpenNMT-kpg-release/blob/master/notebook/json_process.ipynb

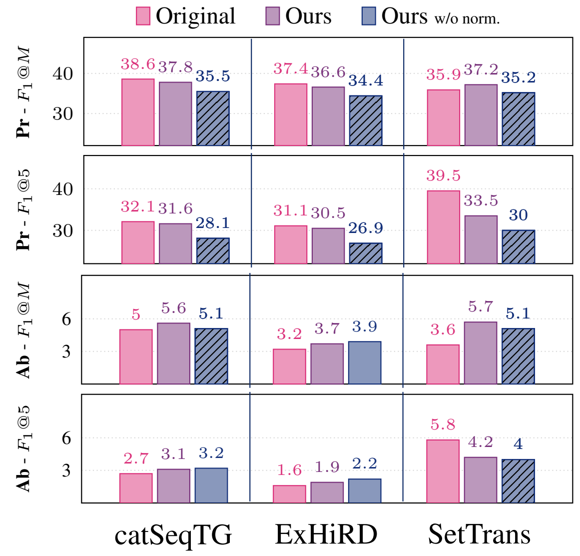

Figure 4: Replicated evaluation results of catSeqTG, ExHiRD and SetTrans on the KP20k dataset, alongside the performance reported in the original paper. Dashed bars (▨) indicate a significant decrease of performance compared to using the normalization, as determined by the Student's paired t-test (p-value < 0.01).

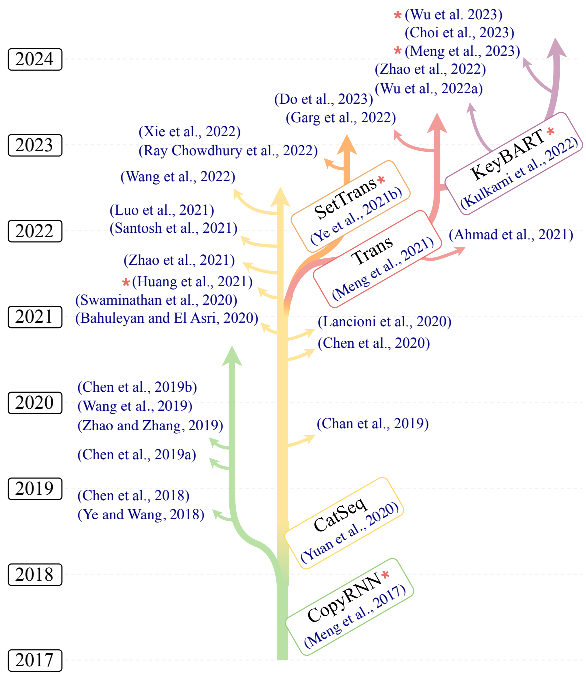

Figure 5: Evolutionary tree of the keyphrase generation models in our survey. Some models are omitted for clarity. * indicate that the model weights are available.

replicated evaluation (■), which we attribute to our method for determining whether a keyphrase is present or absent in the source document. These observations alert that *the performance of many models have been overestimated from using this normalization procedure*, advocating for a cautious comparison of results between studies.

3.3 Proposed Models

Here, we take a closer look at the models proposed in our sample of papers. Figure 5 presents an overview of these models in the form of an evolutionary tree, highlighting five works that we consider important milestones for keyphrase generation. In short, we first witness early efforts dedicated to refining the task formulation of keyphrase generation, followed by a transitional phase from RNN-based to Transformers-based models, and most recently, the adoption of pre-trained language models (PLMs). Below, we provide brief descriptions of each model, organized around these milestone works and presented in chronological order.

2017 Meng et al. (2017) introduced a *RNN-based encoder-decoder model for keyphrase generation*, alongside the KP20k dataset. This model was further improved with additional decoding mechanisms (Chen et al., 2018; Zhao and Zhang, 2019), multi-task learning (Ye and Wang, 2018), external resources (Chen et al., 2019a), latent topic information (Wang et al., 2019; Zhang et al., 2022), better encoding techniques (Chen et al., 2019b; Kim et al., 2021), or self-training (Shen et al., 2022).

2018 Yuan et al. (2020)[3] introduced the ONE2MANY *training paradigm*, enabling models to generate a variable number of keyphrases. Subsequent studies have improved upon this work through the use of reinforcement learning (Chan et al., 2019; Luo et al., 2021), hierarchical decoding (Chen et al., 2020), GANs (Lancioni et al., 2020; Swaminathan et al., 2020), diversity-promoting training objective (Bahuleyan and El Asri, 2020), or diverse decoding strategies (Huang et al., 2021; Zhao et al., 2021; Santosh et al., 2021; Wang et al., 2022).

2021 Meng et al. (2021) explored the generalization capabilities of keyphrase generation models and were among the first to apply *Transformers for this task*. Other works improved the performance of Transformers-based models though manipulation of the input document (Ahmad et al., 2021; Garg et al., 2022) or guided decoding (Do et al., 2023).

[3]This work was submitted to arXiv in October 2018.

2021 Ye et al. (2021b) proposed the ONE2SET *training paradigm* that utilizes control codes to generate a set of keyphrases. Further work improved this work with the use of data augmentation (Ray Chowdhury et al., 2022) or model calibration (Xie et al., 2022).

2022 Kulkarni et al. (2022) investigated the utilization of *PLMs for keyphrase generation*. Subsequent studies confirmed that fine-tuning a PLM, namely BART (Lewis et al., 2020), for keyphrase generation achieves SOTA results (Houbre et al., 2022; Wu et al., 2022a; Meng et al., 2023; Wu et al., 2023), and further improved its performance through output filtering (Zhao et al., 2022), low-resource fine-tuning (Wu et al., 2022a) or contrastive learning (Choi et al., 2023).

Figure 6 provides a more detailed depiction of the architectures (RNN or Transformers) used by the proposed keyphrase generation models over the years. Starting from 2021, we observe a swift transition from RNNs to Transformers, accelerated by the recent line of research on fine-tuning PLMs for the task. This trend aligns with observations across numerous other NLP tasks, where (pre-trained) Transformers consistently achieve state-of-the-art performance.

While it is quite common for previous studies proposing models to release the code for reproducing their experiments (27 out of 38, 71.1%), it is rare for the model weights to be made publicly available, with only 6 out of 38 studies doing so (marked with the symbol $*$ in Figure 5). As shown in previous work, code availability is enough for reproducing the results present in published literature (Arvan et al., 2022). Not having model weights readily available complicates the comparison between models and imposes unnecessary additional computational and environmental costs for retraining. This observation *calls for increased efforts to release model weights*, thereby facilitating further research on keyphrase generation.

3.4 Empirical Results

We conclude our analysis by conducting a large-scale comparison of the performance of the proposed models in our sample of papers, focusing on the best scores they achieve on the KP20k benchmark dataset (see Figure 7). We draw the lines for the state-of-the-art performance over time according to the three most commonly used evaluation metrics for both present and absent keyphrases. Overall, we see a small yet steady increase in state-of-the-art performance, with the latest jump attributed to models leveraging the knowledge of PLMs for the task (Choi et al., 2023; Wu et al., 2023). Two additional observations can be made from the Figure: 1) the absolute improvement in state-of-the-art performance since earlier works is limited; for instance, only 3.5% in present $F_1@M$ separates the works of Chan et al. (2019) and Wu et al. (2023); and 2) the performance in absent keyphrase prediction remains very low, barely reaching 8% in $F_1@M$. We believe that the reasons for this situation could be traced back to the unreliability of the evaluation metrics that rely on strict matching against a single ground truth (see §3.2). This issue becomes more pronounced in the case of absent keyphrases where lexical variation is more prevalent, leading to lower scores.

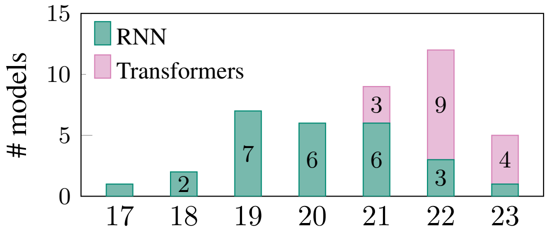

Figure 6: Architectures of the proposed keyphrase generation models over the years.

Another notable observation is the limited use of statistical significance testing in the results of our sampled papers, with only 14 out of the 44 doing so (marked with the symbol • in Figure 7). We assume this is a consequence of the scarce availability of model weights (see §3.3), which hinders the reproducibility of prior research and the ability to directly compare model outputs. Yet, statistical significance testing is crucial to assess the likelihood of potential improvements to models occurring by chance (Dror et al., 2018), casting doubts on the actual progress of the task.

4 A state-of-the-art baseline model

Our analysis offers insights into the progress made by current keyphrase generation models, while also highlighting the lack of uniform evaluation procedures and the limited availability of pre-trained models. Here, we describe our effort to address these issues by building and releasing a state-of-the-art baseline model for keyphrase generation, along with an evaluation framework to facilitate future re-

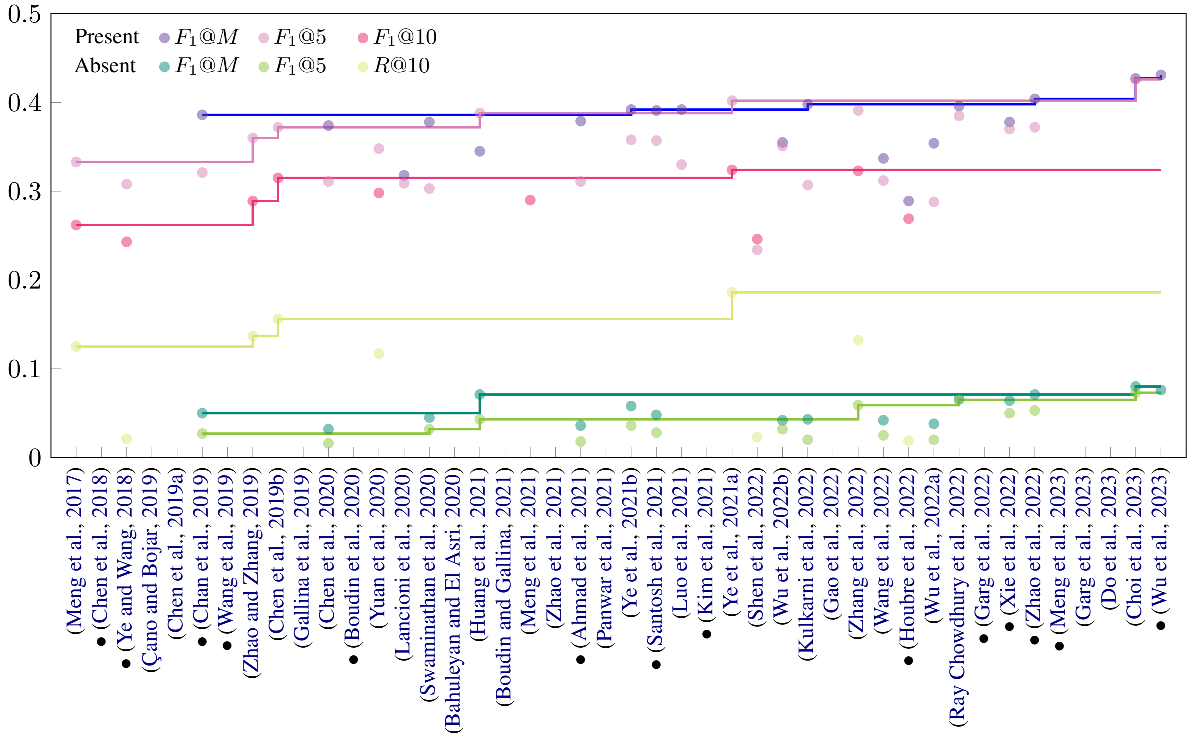

Figure 7: Best scores achieved by each model in terms of $F_1@M$, $F_1@5$ and $F_1@10$ for present keyphrases and $F_1@M$, $F_1@5$ and $R@10$ for absent keyphrases on the KP20k dataset. The lines represent the state-of-the-art performance over time. • indicate that the paper utilizes statistical tests to validate the significance of the results.

search. Upon examining the scores of the proposed models (see §3.4), those that employ fine-tuning a PLM for the task yield the best performance. Accordingly, we adopt this approach for our baseline model and use BART-large (Lewis et al., 2020) as our initial PLM, following (Meng et al., 2023; Wu et al., 2023). We perform fine-tuning on the KP20k training set for 10 epochs in a ONE2MANY setting (Yuan et al., 2020), that is, given a source text as input, the task is to generate keyphrases as a single sequence of delimiter-separated phrases. During fine-tuning, gold keyphrases are arranged in the present-absent order which was found to give the best results (Meng et al., 2021). Implementation details are given in Appendix A.4. It is worth noting that we do not apply any pre-processing to either the source texts or the ground-truth keyphrases, thereby fixing the issues we identified in §3.2. At test time, we use either greedy decoding and let the model generate the most probable keyphrases, or beam search (K=20) and assemble the top-k keyphrases from all the beams as the model output.

To select the best model, we save a checkpoint at the end of each training epoch and evaluate its performance on the validation set of KP20k by calculating the $F_1@\{M, 5, 10\}$ scores against the ground truth keyphrases. Overall, fine-tuning the model for 9 epochs produces the best scores (see Figure 8), leading us to select the corresponding checkpoint as our baseline model. Code for training, inference and evaluation is available at `github.com/anonymous`. Model weights (all checkpoints) are available at `huggingface.co/anonymous`.

Here, we evaluate the performance of our baseline model on the test set of KP20k and see how it compares against previously proposed models. Table 2 presents the results for both present and ab-

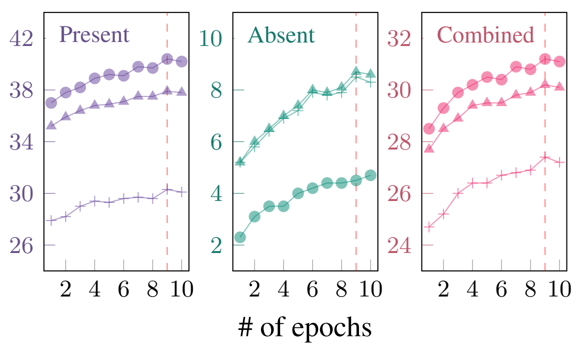

Figure 8: Performance of our baseline model on the validation set of KP20k across each training epoch, measured in terms of $F_1@M$ (○), $F_1@5$ (△) and $F_1@10$ (+) computed for present, absent and combined keyphrases.

sent keyphrase prediction. Overall, we observe that our model achieves strong performance, outperforming most previous models and even achieving state-of-the-art results on absent prediction in terms of $F_1@5$. We believe the performance of our baseline model is sufficiently high to serve as a point of reference in future work, especially considering the potential issue of overestimated performance that we discovered in prior research (see §3.2).

Metric		**Ours**	**Best**	**# ↓**	**# ↑**
$F_1@M$	Present	39.9	43.1	15	3
	Absent	4.5	8.0	4	9
$F_1@5$	Present	37.7	42.6	16	5
	Absent	8.2	7.3	13	0

Table 2: Performance of our baseline model on test set of KP20k, with comparison to the best-reported scores in literature and the number of previous models underperforming (# ↓) or outperforming (# ↑) our baseline.

5 Open Challenges and Discusssion

We wrap up this paper by highlighting two of the open challenges in keyphrase generation and suggesting actionable strategies to address them.

Our analysis revealed alarming levels of redundancy between the most frequently used benchmark datasets, stressing the need to deviate from the common practice of relying solely on the same five datasets. Thus, the first challenge we identified is the *lack of diverse, sizeable benchmark datasets for keyphrase generation*. While recent efforts have been devoted to building new datasets, they either reuse most samples from KP20k (Mahata et al., 2022), contain too few samples (Piedboeuf and Langlais, 2022) or are restricted to a specific domain (Houbre et al., 2022). Creating a new dataset is undoubtedly difficult, as manually annotating keyphrases is costly and necessitates domain experts. One practical solution is to look for naturally occurring keyphrases, and scientific papers with their author-provided keywords are a well-known match. Another common issue of existing datasets is that fall short in sourcing the documents they contain. For instance, documents in KP20k were collected from "various online digital libraries" and lack metadata information such as DOIs, authors or licences. Considering all of the points we mentioned, we suggest leveraging arXiv for creating a new dataset as it aligns with our requirements: it offers content under Creative Commons, provides a substantial volume of categorized, identified and machine-readable (LaTeXand HTML) documents.

The second challenge we identified, which connects to the benchmark datasets, is *the questionable robustness of automatic evaluation*. The main concerns with current evaluation methods are twofold: First, keyphrases are task-dependent. For instance, keyphrases relevant for document indexing may differ from those relevant for reading comprehension. This aspect is hardly ever discussed in previous studies despite its important implications, notably on the need for different ground truth keyphrases depending on the task at hand. One solution to mitigate this issue is to rely on extrinsic evaluation, that is, assessing the performance of keyphrase generation models through downstream tasks. Prior works have, for example, proposed to evaluate models through their impact on document retrieval effectiveness (Boudin et al., 2020; Boudin and Gallina, 2021). However, this methodology has been seldom adopted in current studies, with only one paper implementing it (Wu et al., 2022b). The additional computational costs of conducting such extrinsic evaluation may be responsible for this. Nevertheless, we believe this aspect to be upmost important for grounding the evaluation of the models in the tasks they will be used for. Here, we suggest experimenting with measuring the benefits of adding keyphrases to tasks from existing benchmark datasets, such as SciRepEval (Singh et al., 2023) in the scientific domain or BEIR (Thakur et al., 2021) for heterogeneous retrieval tasks.

Second, commonly-used evaluation metrics rely on exact matching against a single ground truth, which is likely to be incomplete as it is annotated by authors rather than professional indexers. One approach to alleviate this issue is to utilize multiple ground truth annotations, akin to the evaluation methodologies employed in other natural language generation tasks like summarization or machine translation. However, this further increases the costs of an already expensive annotation process, making its adoption unlikely. Another approach to depart from the exact matching evaluation is to leverage semantic information. Recent work explored the use of semantic-based metrics for evaluating generated keyphrases and showed good correlation with human ratings (Wu et al., 2024). Here, we suggest testing the ability of LLMs to evaluate generated keyphrases, as this approach has proven successful in several tasks (Chiang and Lee, 2023).

Limitations

There are two limitations of this paper:

1. While we are confident that the sample of papers covered in this survey represents a comprehensive portion of the research on keyphrase generation, our selection is not exhaustive, disregarding papers from non-ACL journals and pre-print servers.

2. Collecting the best scores from the selected papers was not always possible due to typos or ambiguities in the tables, e.g. out-of-range evaluation scores from Table 5 in (Garg et al., 2023).

References

Wasi Ahmad, Xiao Bai, Soomin Lee, and Kai-Wei Chang. 2021. Select, extract and generate: Neural keyphrase generation with layer-wise coverage attention. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1389–1404, Online. Association for Computational Linguistics.

Mohammad Arvan, Luís Pina, and Natalie Parde. 2022. Reproducibility in computational linguistics: Is source code enough? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2350–2361, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Hareesh Bahuleyan and Layla El Asri. 2020. Diverse keyphrase generation with neural unlikelihood training. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5271–5287, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Florian Boudin and Ygor Gallina. 2021. Redefining absent keyphrases and their effect on retrieval effectiveness. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4185–4193, Online. Association for Computational Linguistics.

Florian Boudin, Ygor Gallina, and Akiko Aizawa. 2020. Keyphrase generation for scientific document retrieval. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1118–1126, Online. Association for Computational Linguistics.

Erion Çano and Ondřej Bojar. 2019. Keyphrase generation: A text summarization struggle. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 666–672, Minneapolis, Minnesota. Association for Computational Linguistics.

Hou Pong Chan, Wang Chen, Lu Wang, and Irwin King. 2019. Neural keyphrase generation via reinforcement learning with adaptive rewards. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2163–2174, Florence, Italy. Association for Computational Linguistics.

Jun Chen, Xiaoming Zhang, Yu Wu, Zhao Yan, and Zhoujun Li. 2018. Keyphrase generation with correlation constraints. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4057–4066, Brussels, Belgium. Association for Computational Linguistics.

Wang Chen, Hou Pong Chan, Piji Li, Lidong Bing, and Irwin King. 2019a. An integrated approach for keyphrase generation via exploring the power of retrieval and extraction. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2846–2856, Minneapolis, Minnesota. Association for Computational Linguistics.

Wang Chen, Hou Pong Chan, Piji Li, and Irwin King. 2020. Exclusive hierarchical decoding for deep keyphrase generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1095–1105, Online. Association for Computational Linguistics.

Wang Chen, Yifan Gao, Jiani Zhang, Irwin King, and Michael R. Lyu. 2019b. Title-guided encoding for keyphrase generation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):6268–6275.

Qikai Cheng, Jiamin Wang, Wei Lu, Yong Huang, and Yi Bu. 2020. Keyword-citation-keyword network: a new perspective of discipline knowledge structure analysis. *Scientometrics*, 124(3):1923–1943.

Ed H. Chi, Michelle Gumbrecht, and Lichan Hong. 2007. Visual foraging of highlighted text: an eye-tracking study. In *Proceedings of the 12th International Conference on Human-Computer Interaction: Intelligent Multimodal Interaction Environments*, HCI'07, page 589–598, Berlin, Heidelberg. Springer-Verlag.

Cheng-Han Chiang and Hung-yi Lee. 2023. A closer look into using large language models for automatic evaluation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 8928–8942, Singapore. Association for Computational Linguistics.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.

Minseok Choi, Chaeheon Gwak, Seho Kim, Si Kim, and Jaegul Choo. 2023. SimCKP: Simple contrastive learning of keyphrase representations. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 3003–3015, Singapore. Association for Computational Linguistics.

Lam Do, Pritom Saha Akash, and Kevin Chen-Chuan Chang. 2023. Unsupervised open-domain keyphrase generation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10614–10627, Toronto, Canada. Association for Computational Linguistics.

Rotem Dror, Gili Baumer, Segev Shlomov, and Roi Reichart. 2018. The hitchhiker's guide to testing statistical significance in natural language processing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1383–1392, Melbourne, Australia. Association for Computational Linguistics.

J. Fagan. 1987. Automatic phrase indexing for document retrieval. In *Proceedings of the 10th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '87, page 91–101, New York, NY, USA. Association for Computing Machinery.

Nazanin Firoozeh, Adeline Nazarenko, Fabrice Alizon, and Béatrice Daille. 2020. Keyword extraction: Issues and methods. *Natural Language Engineering*, 26(3):259–291.

Ygor Gallina, Florian Boudin, and Beatrice Daille. 2019. KPTimes: A large-scale dataset for keyphrase generation on news documents. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 130–135, Tokyo, Japan. Association for Computational Linguistics.

Yifan Gao, Qingyu Yin, Zheng Li, Rui Meng, Tong Zhao, Bing Yin, Irwin King, and Michael Lyu. 2022. Retrieval-augmented multilingual keyphrase generation with retriever-generator iterative training. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1233–1246, Seattle, United States. Association for Computational Linguistics.

Krishna Garg, Jishnu Ray Chowdhury, and Cornelia Caragea. 2022. Keyphrase generation beyond the boundaries of title and abstract. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5809–5821, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Krishna Garg, Jishnu Ray Chowdhury, and Cornelia Caragea. 2023. Data augmentation for low-resource keyphrase generation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8442–8455, Toronto, Canada. Association for Computational Linguistics.

Jiatao Gu, Zhengdong Lu, Hang Li, and Victor O.K. Li. 2016. Incorporating copying mechanism in sequence-to-sequence learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1631–1640, Berlin, Germany. Association for Computational Linguistics.

Carl Gutwin, Gordon Paynter, Ian Witten, Craig Nevill-Manning, and Eibe Frank. 1999. Improving browsing in digital libraries with keyphrase indexes. *Decision Support Systems*, 27(1):81–104.

Kazi Saidul Hasan and Vincent Ng. 2014. Automatic keyphrase extraction: A survey of the state of the art. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1262–1273, Baltimore, Maryland. Association for Computational Linguistics.

Maël Houbre, Florian Boudin, and Beatrice Daille. 2022. A large-scale dataset for biomedical keyphrase generation. In *Proceedings of the 13th International Workshop on Health Text Mining and Information Analysis (LOUHI)*, pages 47–53, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Kai Hu, Qing Luo, Kunlun Qi, Siluo Yang, Jin Mao, Xiaokang Fu, Jie Zheng, Huayi Wu, Ya Guo, and Qibing Zhu. 2019. Understanding the topic evolution of scientific literatures like an evolving city: Using google word2vec model and spatial autocorrelation analysis. *Information Processing & Management*, 56(4):1185–1203.

Xiaoli Huang, Tongge Xu, Lvan Jiao, Yueran Zu, and Youmin Zhang. 2021. Adaptive beam search decoding for discrete keyphrase generation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(14):13082–13089.

Anette Hulth. 2003. Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pages 216–223.

Yi Jiang, Rui Meng, Yong Huang, Wei Lu, and Jiawei Liu. 2023. Generating keyphrases for readers: A controllable keyphrase generation framework. *Journal of the Association for Information Science and Technology*, 74(7):759–774.

Steve Jones and Mark S. Staveley. 1999. Phrasier: A system for interactive document retrieval using keyphrases. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '99, page 160–167, New York, NY, USA. Association for Computing Machinery.

Jihyuk Kim, Myeongho Jeong, Seungtaek Choi, and Seung-won Hwang. 2021. Structure-augmented keyphrase generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2657–2667, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Su Nam Kim, Olena Medelyan, Min-Yen Kan, and Timothy Baldwin. 2010. SemEval-2010 task 5 : Automatic keyphrase extraction from scientific articles. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 21–26, Uppsala, Sweden. Association for Computational Linguistics.

Fajri Koto, Timothy Baldwin, and Jey Han Lau. 2022. LipKey: A large-scale news dataset for absent keyphrases generation and abstractive summarization. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3427–3437, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Mikalai Krapivin, Aliaksandr Autaeu, Maurizio Marchese, et al. 2009. Large dataset for keyphrases extraction. Technical report, University of Trento-Dipartimento di Ingegneria e Scienza dell'Informazione.

Mayank Kulkarni, Debanjan Mahata, Ravneet Arora, and Rajarshi Bhowmik. 2022. Learning rich representation of keyphrases from text. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 891–906, Seattle, United States. Association for Computational Linguistics.

Giuseppe Lancioni, Saida S.Mohamed, Beatrice Portelli, Giuseppe Serra, and Carlo Tasso. 2020. Keyphrase generation with GANs in low-resources scenarios. In *Proceedings of SustaiNLP: Workshop on Simple and Efficient Natural Language Processing*, pages 89–96, Online. Association for Computational Linguistics.

Hwanhee Lee, Seunghyun Yoon, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Joongbo Shin, and Kyomin Jung. 2021. KPQA: A metric for generative question answering using keyphrase weights. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2105–2115, Online. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Yizhu Liu, Qi Jia, and Kenny Zhu. 2021. Keyword-aware abstractive summarization by extracting set-level intermediate summaries. In *Proceedings of the Web Conference 2021*, WWW '21, page 3042–3054, New York, NY, USA. Association for Computing Machinery.

Zhiyuan Liu, Xinxiong Chen, Yabin Zheng, and Maosong Sun. 2011. Automatic keyphrase extraction by bridging vocabulary gap. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning*, pages 135–144, Portland, Oregon, USA. Association for Computational Linguistics.

Wei Lu, Shengzhi Huang, Jinqing Yang, Yi Bu, Qikai Cheng, and Yong Huang. 2021. Detecting research topic trends by author-defined keyword frequency. *Information Processing & Management*, 58(4):102594.

Yichao Luo, Yige Xu, Jiacheng Ye, Xipeng Qiu, and Qi Zhang. 2021. Keyphrase generation with fine-grained evaluation-guided reinforcement learning. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 497–507, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Debanjan Mahata, Navneet Agarwal, Dibya Gautam, Amardeep Kumar, Swapnil Parekh, Yaman Kumar Singla, Anish Acharya, and Rajiv Ratn Shah. 2022. LDKP - A dataset for identifying keyphrases from long scientific documents. In *Proceedings of the Workshop on Deep Learning for Search and Recommendation (DL4SR 2022) co-located with the 31st ACM International Conference on Information and Knowledge Management (CIKM 2022), Atlanta, Georgia, USA, October 17-21, 2022*, volume 3317 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Rui Meng, Tong Wang, Xingdi Yuan, Yingbo Zhou, and Daqing He. 2023. General-to-specific transfer labeling for domain adaptable keyphrase generation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1602–1618, Toronto, Canada. Association for Computational Linguistics.

Rui Meng, Xingdi Yuan, Tong Wang, Sanqiang Zhao, Adam Trischler, and Daqing He. 2021. An empirical study on neural keyphrase generation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4985–5007, Online. Association for Computational Linguistics.

Rui Meng, Sanqiang Zhao, Shuguang Han, Daqing He, Peter Brusilovsky, and Yu Chi. 2017. Deep keyphrase generation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 582–592, Vancouver, Canada. Association for Computational Linguistics.

Thuy Dung Nguyen and Min-Yen Kan. 2007. Keyphrase extraction in scientific publications. In *Asian Digital Libraries. Looking Back 10 Years and Forging New Frontiers*, pages 317–326, Berlin, Heidelberg. Springer Berlin Heidelberg.

Madhur Panwar, Shashank Shailabh, Milan Aggarwal, and Balaji Krishnamurthy. 2021. TAN-NTM: Topic attention networks for neural topic modeling. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3865–3880, Online. Association for Computational Linguistics.

Eirini Papagiannopoulou and Grigorios Tsoumakas. 2020. A review of keyphrase extraction. *WIREs Data Mining and Knowledge Discovery*, 10(2):e1339.

Frédéric Piedboeuf and Philippe Langlais. 2022. A new dataset for multilingual keyphrase generation. In *Advances in Neural Information Processing Systems*, volume 35, pages 38046–38059. Curran Associates, Inc.

M. F. Porter. 1997. *An algorithm for suffix stripping*, page 313–316. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Jishnu Ray Chowdhury, Seo Yeon Park, Tuhin Kundu, and Cornelia Caragea. 2022. KPDROP: Improving absent keyphrase generation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4853–4870, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Anna Rogers, Marzena Karpinska, Jordan Boyd-Graber, and Naoaki Okazaki. 2023. Program chairs' report on peer review at acl 2023. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages xl–lxxv, Toronto, Canada. Association for Computational Linguistics.

Tokala Yaswanth Sri Sai Santosh, Nikhil Reddy Varimalla, Anoop Vallabhajosyula, Debarshi Kumar Sanyal, and Partha Pratim Das. 2021. Hicova: Hierarchical conditional variational autoencoder for keyphrase generation. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, CIKM '21, page 3448–3452, New York, NY, USA. Association for Computing Machinery.

Xianjie Shen, Yinghan Wang, Rui Meng, and Jingbo Shang. 2022. Unsupervised deep keyphrase generation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(10):11303–11311.

Amanpreet Singh, Mike D'Arcy, Arman Cohan, Doug Downey, and Sergey Feldman. 2023. SciRepEval: A multi-format benchmark for scientific document representations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5548–5566, Singapore. Association for Computational Linguistics.

Mingyang Song, Yi Feng, and Liping Jing. 2023. A survey on recent advances in keyphrase extraction from pre-trained language models. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 2153–2164, Dubrovnik, Croatia. Association for Computational Linguistics.

Sandeep Subramanian, Tong Wang, Xingdi Yuan, Saizheng Zhang, Adam Trischler, and Yoshua Bengio. 2018. Neural models for key phrase extraction and question generation. In *Proceedings of the Workshop on Machine Reading for Question Answering*, pages 78–88, Melbourne, Australia. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc.

Avinash Swaminathan, Haimin Zhang, Debanjan Mahata, Rakesh Gosangi, Rajiv Ratn Shah, and Amanda Stent. 2020. A preliminary exploration of GANs for keyphrase generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8021–8030, Online. Association for Computational Linguistics.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. Beir: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1.

Xiaojun Wan and Jianguo Xiao. 2008. Single document keyphrase extraction using neighborhood knowledge. In *Proceedings of the 23rd National Conference on Artificial Intelligence - Volume 2*, AAAI'08, page 855–860. AAAI Press.

Xiaojun Wan, Jianwu Yang, and Jianguo Xiao. 2007. Towards an iterative reinforcement approach for simultaneous document summarization and keyword extraction. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 552–559, Prague, Czech Republic. Association for Computational Linguistics.

Siyu Wang, Jianhui Jiang, Yao Huang, and Yin Wang. 2022. Automatic keyphrase generation by incorporating dual copy mechanisms in sequence-to-sequence learning. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2328–2338, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Yue Wang, Jing Li, Hou Pong Chan, Irwin King, Michael R. Lyu, and Shuming Shi. 2019. Topic-aware neural keyphrase generation for social media language. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2516–2526, Florence, Italy. Association for Computational Linguistics.

Di Wu, Wasi Ahmad, and Kai-Wei Chang. 2023. Rethinking model selection and decoding for keyphrase generation with pre-trained sequence-to-sequence models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6642–6658, Singapore. Association for Computational Linguistics.

Di Wu, Wasi Ahmad, Sunipa Dev, and Kai-Wei Chang. 2022a. Representation learning for resource-constrained keyphrase generation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 700–716, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Di Wu, Da Yin, and Kai-Wei Chang. 2024. Kpeval: Towards fine-grained semantic-based keyphrase evaluation.

Huanqin Wu, Baijiaxin Ma, Wei Liu, Tao Chen, and Dan Nie. 2022b. Fast and constrained absent keyphrase generation by prompt-based learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(10):11495–11503.

Binbin Xie, Jia Song, Liangying Shao, Suhang Wu, Xiangpeng Wei, Baosong Yang, Huan Lin, Jun Xie, and Jinsong Su. 2023. From statistical methods to deep learning, automatic keyphrase prediction: A survey. *Information Processing & Management*, 60(4):103382.

Binbin Xie, Xiangpeng Wei, Baosong Yang, Huan Lin, Jun Xie, Xiaoli Wang, Min Zhang, and Jinsong Su. 2022. WR-One2Set: Towards well-calibrated keyphrase generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7283–7293, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jianxin Yang, Wenge Rong, Libin Shi, and Zhang Xiong. 2019. Sequential Attention with Keyword Mask Model for Community-based Question Answering. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2201–2211, Minneapolis, Minnesota. Association for Computational Linguistics.

Hai Ye and Lu Wang. 2018. Semi-supervised learning for neural keyphrase generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4142–4153, Brussels, Belgium. Association for Computational Linguistics.

Jiacheng Ye, Ruijian Cai, Tao Gui, and Qi Zhang. 2021a. Heterogeneous graph neural networks for keyphrase generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2705–2715, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jiacheng Ye, Tao Gui, Yichao Luo, Yige Xu, and Qi Zhang. 2021b. One2Set: Generating diverse keyphrases as a set. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4598–4608, Online. Association for Computational Linguistics.

Xingdi Yuan, Tong Wang, Rui Meng, Khushboo Thaker, Peter Brusilovsky, Daqing He, and Adam Trischler. 2020. One size does not fit all: Generating and evaluating variable number of keyphrases. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7961–7975, Online. Association for Computational Linguistics.

Hongyuan Zha. 2002. Generic summarization and keyphrase extraction using mutual reinforcement principle and sentence clustering. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '02, page 113–120, New York, NY, USA. Association for Computing Machinery.

Chengxiang Zhai. 1997. Fast statistical parsing of noun phrases for document indexing. In *Fifth Conference on Applied Natural Language Processing*, pages 312–319, Washington, DC, USA. Association for Computational Linguistics.

Yuxiang Zhang, Tao Jiang, Tianyu Yang, Xiaoli Li, and Suge Wang. 2022. Htkg: Deep keyphrase generation with neural hierarchical topic guidance. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, page 1044–1054, New York, NY, USA. Association for Computing Machinery.

Guangzhen Zhao, Guoshun Yin, Peng Yang, and Yu Yao. 2022. Keyphrase generation via soft and hard semantic corrections. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7757–7768, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jing Zhao, Junwei Bao, Yifan Wang, Youzheng Wu, Xiaodong He, and Bowen Zhou. 2021. SGG: Learning to select, guide, and generate for keyphrase generation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5717–5726, Online. Association for Computational Linguistics.

Jing Zhao and Yuxiang Zhang. 2019. Incorporating linguistic constraints into keyphrase generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5224–5233, Florence, Italy. Association for Computational Linguistics.

Erion Çano and Ondřej Bojar. 2019. Keyphrase generation: A multi-aspect survey. In *2019 25th Conference of Open Innovations Association (FRUCT)*, pages 85–94.

A Appendix

A.1 Statistics of the Benchmark Datasets

Detailed statistics of the datasets are provided in Table 4.

A.2 Details of Evaluation Metrics

For a given document d, the performance of a model is evaluated by comparing its predicted keyphrases $\mathcal{P} = \{p_1, p_2, \cdots, p_M\}$ with a set of gold truth keyphrases $\mathcal{Y} = \{y_1, y_2, \cdots, y_O\}$. Keyphrases are lowercased, stemmed with the Porter Stemmer (Porter, 1997), and duplicates are removed prior to score calculation. When only the top-k predictions $\mathcal{P}_{:k} = \{p_1, \cdots, p_{\min(k,M)}\}$ are used for evaluation, the *precision*, *recall* and F_1 *measure* are computed as follows:

$$P@k = \frac{|\mathcal{P}_{:k} \cap \mathcal{Y}|}{|\mathcal{P}_{:k}|} \quad R@k = \frac{|\mathcal{P}_{:k} \cap \mathcal{Y}|}{|\mathcal{Y}|}$$

$$F_1@k = 2 \times \frac{P@k \times R@k}{P@k + R@k}$$

The most commonly used metrics are defined as:

- $F_1@5$: $F_1@k$ when $k = 5$.
- $F_1@10$: $F_1@k$ when $k = 10$.
- $F_1@M$: M denotes the number of predicted keyphrases. Here, all the predicted phrases are used for evaluation, i.e. without truncation.
- $F_1@O$: O denotes the number of gold truth keyphrases.
- $R@10$: $R1@k$ when $k = 10$.
- $R@50$: $R1@k$ when $k = 50$.

Noting that when using the top-k predictions and the number of predicted keyphrases M is lower than k, incorrect phrases are appended to $\mathcal{P}$ until that M reaches k.

A keyphrase is labelled as present if it constitutes a subsequence of token of d (in stemmed form), and absent otherwise. When results for present and absent are reported separately, only the present or absent keyphrases from $\mathcal{P}$ and Y and used for score calculation. Papers usually report the macro-average scores over all the data examples in a benchmark dataset.

A.3 Example of normalized keyphrases

An example of data normalization as in Meng et al. (2017) is presented in Table 3.

Title: Autoimmune polyendocrinopathy candidiasis ectodermal dystrophy: known and novel aspects of the syndrome
Abstract: Autoimmune polyendocrinopathy candidiasis ectodermal dystrophy (APECED) is a monogenic autosomal recessive disease caused by mutations in the autoimmune regulator (AIRE) gene and, as a syndrome, is characterized by chronic mucocutaneous candidiasis and the presentation of various autoimmune diseases. During the last decade, research on APECED and AIRE has provided immunologists with several invaluable lessons regarding tolerance and autoimmunity. This review describes the clinical and immunological features of APECED and discusses emerging alternative models to explain the pathogenesis of the disease.
Keyphrases: apeced – aire – chronic mucocutaneous candidiasis – il-17 – il-22 **Normalized**: apeced – aire – chronic mucocutaneous candidiasis – il <digit>

Table 3: Example of document from KP20k (S2CID: 32645143) with its associated keyphrases and their normalized forms.

A.4 Implementation Details

We use the BART-large model weights as our initial pre-trained language model and perform fine-tuning on the KP20k training set for 10 epochs. We use the AdamW optimizer with a learning rate of 1e-5 and a batch size of 4. Fine-tuning the model using 2 Nvidia GeForce RTX 2080 took 400 hours.

Dataset	train / dev / test	#kp	|kp|	%abs
KP20k (Meng et al., 2017)	514k / 20k / 20k	5.3	2.1	36.7
SemEval-2010 (Kim et al., 2010)	144 / – / 100	15.7	2.1	55.5
Inspec (Hulth, 2003)	1k / 500 / 500	9.6	2.3	21.5
Krapivin (Krapivin et al., 2009)	1844 / - / 460	5.2	2.2	43.8
NUS (Nguyen and Kan, 2007)	– / – / 211	11.5	2.2	48.7
DUC2001 (Wan and Xiao, 2008)	– / – / 308	8.1	2.1	2.7
KPTimes (Gallina et al., 2019)	260k / 10k / 20k	5.0	1.5	54.4
StackEx (Yuan et al., 2020)	298k / 16k / 16k	2.7	–	42.5
Weibo (Wang et al., 2019)	37k / 4.6k / 4.6k	1.1	2.6	75.8
StackEx (Wang et al., 2019)	39.6k / 4.9k / 4.9k	2.4	1.4	54.3

Table 4: Statistics of the benchmark datasets taken from (Wan and Xiao, 2008; Gallina et al., 2019; Wang et al., 2019; Yuan et al., 2020; Do et al., 2023)