An Analysis of Datasets, Metrics and Models in 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. In this paper, we bridge this gap by presenting an analysis of over 50 research papers on keyphrase generation, offering a comprehensive overview of recent progress, limitations, and open challenges. Our findings highlight several critical issues in current evaluation practices, such as the concerning similarity among commonly-used benchmark datasets and inconsistencies in metric calculations leading to overestimated performances. Additionally, we address the limited availability of pre-trained models by releasing a strong PLM-based model for keyphrase generation 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., 2023a).

Keyphrase generation expands on keyphrase extraction by enabling the production of *keyphrases absent from the source text* (Liu et al., 2011). This ability is critical when dealing with short documents that often lack appropriate keyphrase candidates. Meng et al. (2017) provided the seminal work on keyphrase generation, introducing a sequence-to-sequence learning approach. Their model builds upon an RNN encoder-decoder architecture (Cho et al., 2014; Sutskever et al., 2014) and incorporates a copying mechanism (Gu et al., 2016) to identify important phrases within the source text. Equally importantly, they introduced KP20k, a dataset that laid the groundwork for end-to-end training of neural models for keyphrase generation.

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 analysis paper 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 50 papers on keyphrase generation, identifying the type(s) of contribution these papers made (§3.1), examining the most frequently used benchmark datasets (§3.2) and evaluation metrics (§3.3), providing descriptions of proposed models while highlighting important milestones (§3.4), and investigating how proposed models perform against each other (§3.5).

Our findings are that: 1) commonly used benchmark datasets are so similar that reporting results on more than one adds no value, 2) the performance of models is often overestimated due to discrepancies in evaluation protocols, and 3) while dedicated models have been superseded by fine-tuned pre-trained language models (PLMs), the overall performance gain since early models remains limited.

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 Scope of the Study

Our analysis encompasses a total of 52 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 seminal 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 venues. 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 826 triples from our sample, corresponding to 50 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`.

3 Analysis

In this section, we analyze the selected papers across five key dimensions: types of contribution (§3.1), benchmark datasets (§3.2), evaluation metrics (§3.3), model architectures (§3.4), and the best reported performances (§3.5).

3.1 Types of contribution

We start our analysis by presenting statistics on the types of contribution made in the papers we examined (see Table 1). Most of the papers propose new models (87%), suggesting that the primary emphasis within the field is on improving the performance of the state-of-the-art. This trend is reinforced by the fact that the second most common contribution is data resources (19%), essential for validating improvements. Some attention was given to model analysis and interpretability (14%), 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). Approaches for low-resource settings also received some attention (14%), initially with data-efficient models (Lancioni et al., 2020; Wu et al., 2022a), then through data augmentation methods (Gao et al., 2022; Garg et al., 2023; Kang and Shin, 2024) and most recently, domain adaptation strategies (Boudin and Aizawa, 2024).

One underexplored area is the development of low-compute approaches, maybe overlooked in the race toward larger models designed to boost performance. This trend contrasts with practical applications, such as document indexing, where speed and efficiency are critical. Our analysis also reveals the limited attention given to data analysis, repro-

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

duction studies and surveys in the literature.[2] This paper seeks to address this gap by providing new insights into the redundancy of existing datasets, conducting replication experiments on model evaluation, and offering a comprehensive overview of models for keyphrase generation.

Type of contribution	%
NLP engineering experiment	86.5
Data resources	19.2
Model analysis and interpretability	13.5
Approaches for low-resource settings	13.5
Software and pre-trained models	7.7
Reproduction studies	1.9

Table 1: Percentage of papers making each type of contribution (a paper may contribute to multiple types).

3.2 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.2). We find that 26 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, with 22 out of 52 papers (42%) employing all five, and 39 out of 52 (75%) employing at least two. All five datasets exclusively contain scientific abstracts, whereas the remaining datasets are sourced from various domains, such as news, social media and web pages. This domain bias can be attributed to two main factors: the availability of scientific abstracts, and the frequent presence of author-assigned keyphrases, 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. The majority of these datasets, however, are relatively small (i.e. $< 1k$ samples) and are mainly used for testing purposes.

A closer examination of the five widely-used datasets reveals substantial overlap. All consist of scientific abstracts from the Computer Science domain, and at least three—KP20k, SemEval-2010, and Krapivin—share documents from the same

[2]We note, however, that several surveys on keyphrase extraction have been conducted; see Appendix A.1 for a review.

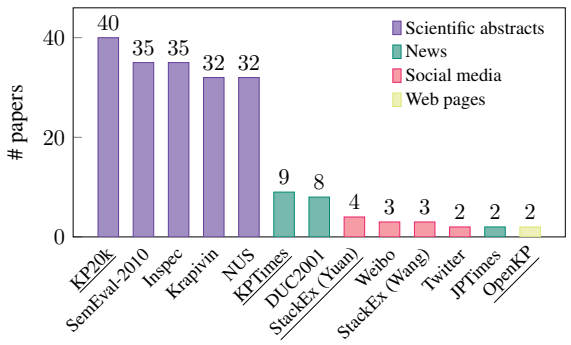

Figure 1: Number of papers utilizing each dataset. Underlined datasets contain $100k$+ training samples. Datasets used only once are omitted for clarity.

source, the ACM Digital Library. This raises concerns about potential data leakage and questions the value of using these datasets together in experimental setups.

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.

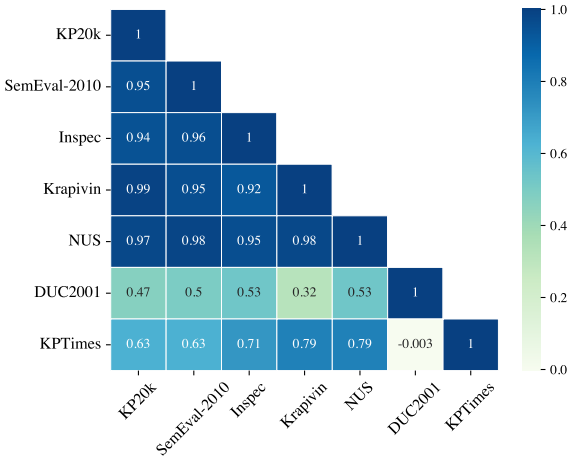

Figure 2: Pearson's correlation coefficient ρ computed between the model scores across datasets.

3.3 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, typically using exact matching. From the extracted score triples, we find that 42 distinct evaluation metrics were reported across the papers (see Figure 3, detailed definitions of the evaluation metrics are provided in §A.3). The majority of papers describing models (40 out of 50, 80%) 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, 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 $k \in \{5, 10\}$).

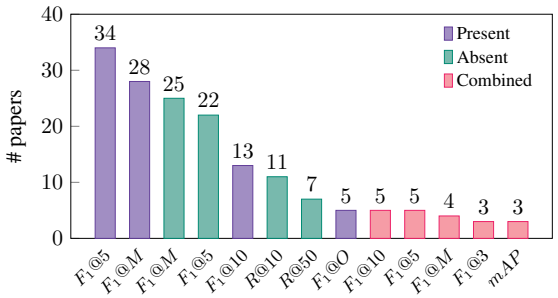

Figure 3: Number of papers employing each evaluation metric. Metrics used < 3 times are excluded for clarity.

Upon closer inspection of the evaluation settings in our sample of papers, we identified two major inconsistencies in how metrics are calculated. First, two variants of $F_1@k$ co-exist. Starting with (Chan et al., 2019), model predictions that do not reach k keyphrases are extended with incorrect (dummy) phrases. This prevents $F_1@k$ and $F_1@M$ scores from being identical, but lowers the scores for models generating fewer than k keyphrases. More critically, this practice undermines direct comparability with earlier work.

Second, we find that some form of normalization procedure is frequently applied prior to computing evaluation metrics, as observed in at least 30 out of 50 papers (60%).[3] This procedure, commonly referred to as Meng et al. (2017)'s pre-processing, 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.4), 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. To ensure comparability and consistency, we compute $F_1@k$ scores with dummy phrases when the number of predicted keyphrases is less than k, following (Chan et al., 2019) and subsequent works.

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

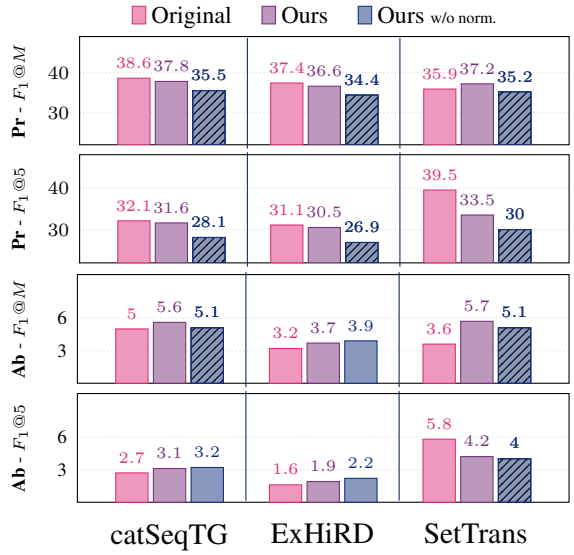

Figure 4: Replicated evaluation results on the KP20k dataset, alongside the performance reported in the original paper. Dashed bars ( ) indicate a significant decrease of performance compared to normalization, as determined by the Student's paired t-test (p-value < 0.01).

[3]This information is often difficult to locate, as it is frequently omitted from papers and requires examining the source code and data.

replicated evaluation ( ), which we attribute to our method for determining whether a keyphrase is present or absent in the source document (see A.3). 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.

3.4 Proposed Models

In this section, 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 fine-tuned 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) introduced the ONE2MANY *training paradigm*, enabling models to generate a variable number of keyphrases.[4] 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 promotion (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).

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

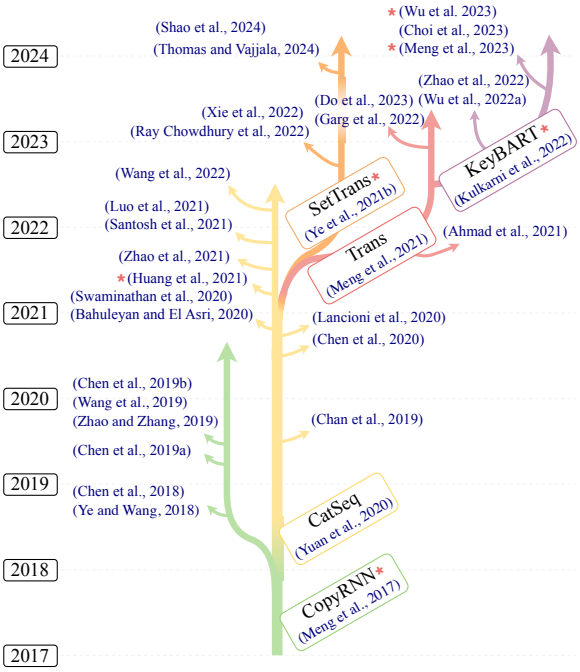

Figure 5: Evolutionary tree of the keyphrase generation models in our analysis. Some models are omitted for clarity. * indicate that the model weights are available.

2021 Ye et al. (2021b) proposed the ONE2SET *training paradigm* that utilizes control codes to generate a set of keyphrases. Further work improved this approach through data augmentation (Ray Chowdhury et al., 2022), model calibration (Xie et al., 2022), joint keyphrase extraction (Thomas and Vajjala, 2024a) or LLM verification (Shao et al., 2024).

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 (Wu et al., 2021; 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; Kang and Shin, 2024; Boudin and Aizawa, 2024), contrastive learning (Choi et al., 2023) or encoder-only models (Wu et al., 2024a).

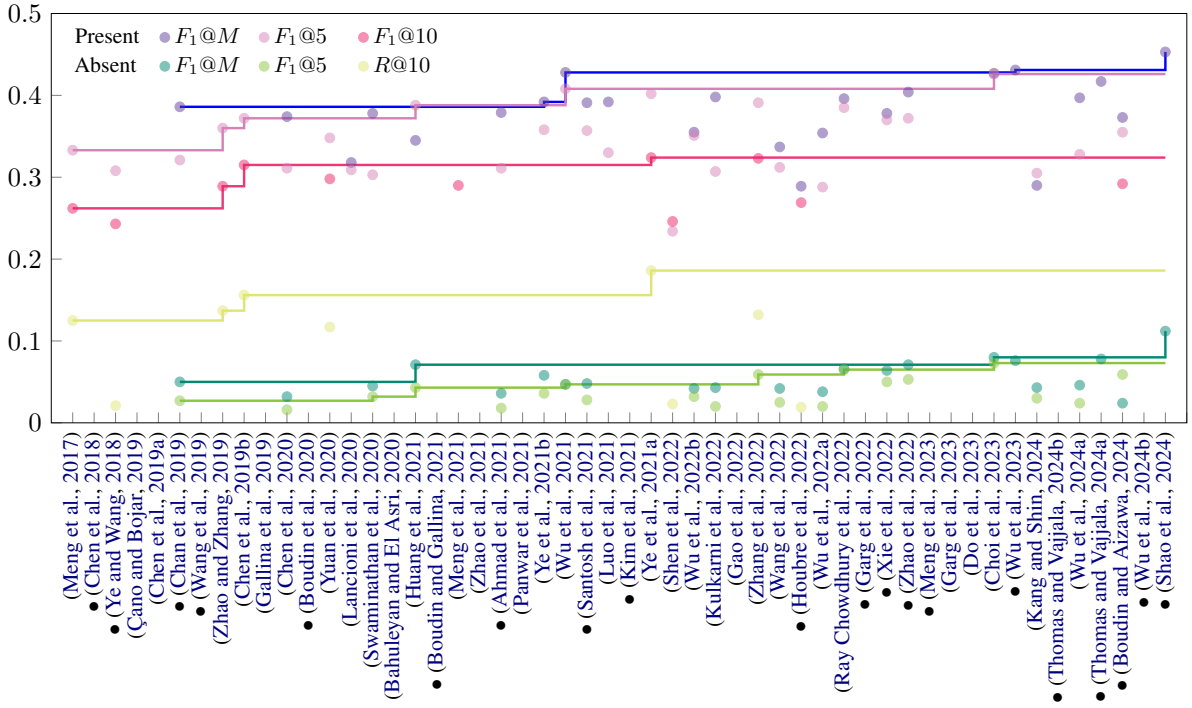

Figure 6: Best scores achieved by each model in terms of $F_1@M$, $F_1@5$ and $F_1@10$ for present keyphrases and $F_1@M$, $F_1@5$ and $R@10$ for absent keyphrases on the KP20k dataset. The lines represent the state-of-the-art performance over time. • indicate that the paper utilizes statistical tests to validate the significance of the results.

Figure 7 provides a more detailed depiction of the architectures 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.

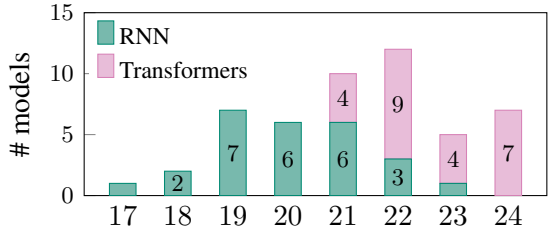

Figure 7: Architectures of the proposed keyphrase generation models over the years.

While it is relatively common for studies introducing models to release the code for reproducing their experiments (34 out of 50, 68%), it is much rarer for the model weights to be made available, with only 8 out of 50 studies doing so (marked with the symbol $*$ in Figure 5). Importantly, code availability alone is not enough for reproducing the results reported in published literature (Arvan et al., 2022). This lack of model weights complicates fair comparisons between models and imposes unnecessary computational and environmental costs associated with retraining.

3.5 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 6). We plot the state-of-the-art performance over time using lines, considering the three most commonly used evaluation metrics for both present and absent keyphrases. To the best of our knowledge, this is the comprehensive overview of state-of-the-art keyphrase generation model performance over time.

Overall, we observe a modest yet steady increase in state-of-the-art performance, with the most recent leap attributed to the use of LLMs for filtering keyphrase candidates generated by a fine-tuned PLM (Shao et al., 2024). Two additional observations can be gleaned from the Figure: 1) the absolute improvement in state-of-the-art performance since earlier works is limited; for instance, only 3.1% in present $F_1@M$ separates the

works of Chan et al. (2019) and Thomas and Vajjala (2024a); and 2) the performance in absent keyphrase prediction remains notably low, barely reaching 11% in $F_1@M$. For the latter, we believe that the reasons could be traced back to the unreliability of the evaluation metrics, which rely on strict matching against a single ground truth (see §3.3). This issue becomes more pronounced in the case of absent keyphrases where lexical variation is more prevalent, leading to lower scores.

Another notable observation is the limited use of statistical significance testing in the results, with only 20 out of the 50 doing so (marked with • in Figure 6). We assume this is a consequence of the scarce availability of model weights (see §3.4), 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 strong baseline model

Our analysis highlights the progress achieved by current keyphrase generation models, while drawing attention to the lack of standardized evaluation procedures and the limited availability of pretrained models. To address these challenges, we provide a strong baseline model for keyphrase generation, along with an evaluation framework to facilitate future research.

Upon analysing the proposed model scores (see §3.5), we find that fine-tuning a PLM for the task consistently yields the best performance. Based on this observation, we adopt this approach for our baseline model, leveraging BART-large (Lewis et al., 2020) as the initial PLM, in line with recent studies (Meng et al., 2023; Wu et al., 2023). The model is fine-tuned 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). Notably, we do not apply any pre-processing to either the source texts or the ground-truth keyphrases, thereby fixing the issues we identified in §3.3.

At test time, we evaluate the model using greedy decoding to generate the most probable keyphrases, or beam search (K=20) to assemble the top-k keyphrases across all beams. To select the best-performing model, we save a checkpoint at the end of each epoch and evaluate its performance on the validation set of KP20k, using $F_1@\{M, 5, 10\}$ scores against the ground truth keyphrases. Overall, fine-tuning for 9 epochs produces the highest 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`. Additionally, all model weights, including checkpoints, are accessible at `huggingface.co/anonymous`. Implementation details are given in Appendix A.5.

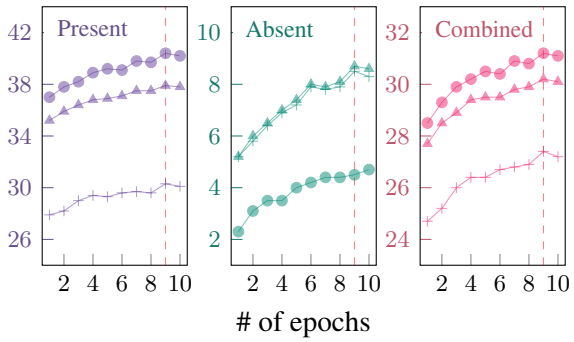

Figure 8: Performance of our baseline model on the KP20k validation set across training epochs, measured by $F_1@M$ (○), $F_1@5$ (△) and $F_1@10$ (+) for present, absent and combined keyphrases.

Here, we evaluate the performance of our baseline model on KP20k test set and compare it against previously proposed models. Table 2 summarizes the results for both present and absent keyphrase prediction. Our model achieves strong overall performance, surpassing most prior models and achieving state-of-the-art results in absent keyphrase prediction in terms of $F_1@5$. We believe that this level of performance establishes our baseline model as a robust point of reference for future research.

Metric		Ours	Best	# ↓	# ↑
$F_1@M$	Present	39.9	45.3	19	6
	Absent	4.5	11.2	9	13
$F_1@5$	Present	37.7	42.6	19	6
	Absent	8.2	7.3	23	0

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

5 Open Challenges and Discussion

We wrap up this paper by highlighting two challenges in keyphrase generation and suggesting actionable strategies to address them. Finally, we discuss what LLMs can do for the task.

5.1 Benchmark Datasets

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 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 domains (Houbre et al., 2022; Boudin and Aizawa, 2024) or goals (Wu et al., 2024b).

Creating a new dataset is undoubtedly difficult, as manual annotation of keyphrases is both costly and requires domain expertise. A 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 their lack of proper document sourcing. For instance, the documents in KP20k were collected from "various online digital libraries" and lack crucial metadata such as DOIs, authorship details or licences. Given these considerations, 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 (LaTeX) documents.

5.2 Evaluation Metrics

The second challenge we identified, which connects to the benchmark datasets, concerns the questionable robustness of automatic evaluation. There are two main issues with current evaluation methods. First, keyphrases are task-dependent. For example, keyphrases relevant for document indexing may differ from those relevant for reading comprehension. This aspect is rarely addressed in previous studies, despite its significant implications, notably on the need for distinct ground truth keyphrases depending on the targeted task. Second, commonly-used evaluation metrics rely on simple matching against a single ground truth, which is likely to be incomplete.

One potential solution to address these issues is to rely on extrinsic evaluation, that is, assessing the performance of keyphrase generation models through downstream tasks. For instance, prior works have proposed to evaluate models through their impact on document retrieval effectiveness (Boudin et al., 2020; Boudin and Gallina, 2021). Two other notable works in this direction are Jiang et al. (2023b), which evaluates keyphrases in a task-oriented setting to assist reader comprehension, and Wu et al. (2024c), which examines the alignment between keyphrases and LLM-generated queries. However, the additional computational costs associated with conducting such extrinsic evaluations may hinder their adaption. 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).

5.3 LLMs for Keyphrase Generation

Keyphrase generation stands out as one of the few NLP tasks where LLMs have not yet replaced dedicated supervised models. Nonetheless, initial efforts to leverage LLMs for this task, primarily using in-context learning (Song et al., 2023b; Martínez-Cruz et al., 2023; Bai et al., 2024), have demonstrated promising results. Recently, Shao et al. (2024) validated the effectiveness of LLMs as a keyphrase reranking method for dedicated models. Here, we highlight two important considerations when using LLMs for keyphrase generation.

The first is data contamination, which occurs when test data is included in the model's training data. Given the extensive size and diverse sources of pre-training datasets used for LLMs, it is likely that widely available documents composing the current benchmarks have been included. Solutions to address this issue are not straightforward, but applying pre-training data detection methods (Zhou et al., 2024; Zhang et al., 2024) to identify and mitigate data leakage is a necessary first step.

The second is the computational costs. Generating keyphrases using LLMs across a vast collection of documents is prohibitively expensive. While "lightweight" models (Grattafiori et al., 2024) or fast inference strategies (Liu et al., 2024) are being developed to reduce these costs, scalable solutions remain an open challenge. Reporting the performance-inference speed trade-off of future models would help better position their practical usefulness.

Limitations

Scope of the analysis

While we are confident that the sample of papers covered in this analysis provides a comprehensive representation of the research on keyphrase generation, our selection is not exhaustive. Specifically, it does not account for papers published in non-ACL journals or hosted on pre-print servers, which may present additional perspectives or recent advancements in the field. Our analysis focuses on keyphrase generation and does not cover the closely related field of keyphrase extraction, which converges on the datasets and evaluation metrics.

Manual extraction of best scores

Our analysis focuses on the best scores reported for the models and could be extended to include baselines and ablation studies. Collecting the best scores from the selected papers was not always possible due to typos or ambiguities in the tables. Furthermore, our disambiguation strategy—selecting either the model demonstrating the best overall performance or, when unclear, the one performing best on the KP20k dataset—may result in suboptimal scores for other datasets.

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.

Xiao Bai, Xue Wu, Ivan Stojkovic, and Kostas Tsioutsiouliklis. 2024. Leveraging large language models for improving keyphrase generation for contextual targeting. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, CIKM '24, page 4349–4357, New York, NY, USA. Association for Computing Machinery.

Florian Boudin and Akiko Aizawa. 2024. Unsupervised domain adaptation for keyphrase generation using citation contexts. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 598–614, Miami, Florida, USA. Association for 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.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park,

Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal

Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. 2024. The llama 3 herd of models.

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. 2023a. Generating keyphrases for readers: A controllable keyphrase generation framework. *Journal of the Association for Information Science and Technology*, 74(7):759–774.

Yi Jiang, Rui Meng, Yong Huang, Wei Lu, and Jiawei Liu. 2023b. 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.

Byungha Kang and Youhyun Shin. 2024. 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)*, pages 8853–8865, Torino, Italia. ELRA and ICCL.

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.

Jiahao Liu, Qifan Wang, Jingang Wang, and Xunliang Cai. 2024. Speculative decoding via early-exiting for faster LLM inference with Thompson sampling control mechanism. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 3027–3043, Bangkok, Thailand. 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.

Roberto Martínez-Cruz, Alvaro J. López-López, and José Portela. 2023. Chatgpt vs state-of-the-art models: A benchmarking study in keyphrase generation task.

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.

Liangying Shao, Liang Zhang, Minlong Peng, Guoqi Ma, Hao Yue, Mingming Sun, and Jinsong Su. 2024. One2Set + large language model: Best partners for keyphrase generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 11140–11153, Miami, Florida, USA. Association for Computational Linguistics.

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.

Mingyang Song, Yi Feng, and Liping Jing. 2023a. 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.

Mingyang Song, Haiyun Jiang, Shuming Shi, Songfang Yao, Shilong Lu, Yi Feng, Huafeng Liu, and Liping Jing. 2023b. Is chatgpt a good keyphrase generator? a preliminary study.

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.

Edwin Thomas and Sowmya Vajjala. 2024a. Improving absent keyphrase generation with diversity heads. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1568–1584, Mexico City, Mexico. Association for Computational Linguistics.

Edwin Thomas and Sowmya Vajjala. 2024b. Keyphrase generation: Lessons from a reproducibility study. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 9720–9731, Torino, Italia. ELRA and ICCL.

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, and Kai-Wei Chang. 2024a. On leveraging encoder-only pre-trained language models for effective keyphrase generation. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 12370–12384, Torino, Italia. ELRA and ICCL.

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, Xiaoxian Shen, and Kai-Wei Chang. 2024b. MetaKP: On-demand keyphrase generation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 8420–8437, Miami, Florida, USA. Association for Computational Linguistics.

Di Wu, Da Yin, and Kai-Wei Chang. 2024c. KPEval: Towards fine-grained semantic-based keyphrase evaluation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 1959–1981, Bangkok, Thailand. Association for Computational Linguistics.

Huanqin Wu, Wei Liu, Lei Li, Dan Nie, Tao Chen, Feng Zhang, and Di Wang. 2021. UniKeyphrase: A unified extraction and generation framework for keyphrase prediction. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 825–835, Online. Association for Computational Linguistics.

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.

Weichao Zhang, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Yixing Fan, and Xueqi Cheng. 2024. Pretraining data detection for large language models: A divergence-based calibration method. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5263–5274, Miami, Florida, 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.

Baohang Zhou, Zezhong Wang, Lingzhi Wang, Hongru Wang, Ying Zhang, Kehui Song, Xuhui Sui, and Kam-Fai Wong. 2024. DPDLLM: A black-box framework for detecting pre-training data from large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 644–653, Bangkok, Thailand. 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 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. (2023a) 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.

A.2 Statistics of the Benchmark Datasets

Detailed statistics of the datasets are provided in Table 4.

A.3 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. This method is stricter than regex-based matching commonly used in previous work. 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.4 Example of normalized keyphrases

An example of data normalization as in Meng et al. (2017)[5] is presented in Table 3.

A.5 Implementation Details

We use the BART-large model weights[6] as our initial pre-trained language model and perform fine-tuning on the KP20k training set[7] 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.

[5] https://github.com/memray/OpenNMT-kpg-release/blob/d16bf09e21521a6854ff3c7fe6eb271412914960/notebook/json_process.ipynb

[6] https://huggingface.co/facebook/bart-large

[7] https://huggingface.co/datasets/taln-ls2n/kp20k

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.

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)