MDPI

Article

Evaluating AI-generated text summaries using Open AI's GPT model

Hassan Shakil [1,], Atqiya Munawara Mahi [2,], Phuoc Nguyen [3,], Zeydy Ortiz [4,] and Mamoun T. Mardini [5,*]

[1] University of Colorado - Colorado Springs
[2] University of Massachusetts Lowell
[3] University of Kansas, US Army
[4] DataCrunch Lab, LLC
[5] Department of Health Outcomes and Biomedical Informatics, University of Florida; malmardini@ufl.edu
* Correspondence: malmardini@ufl.edu
‡ These authors contributed equally to this work.

Abstract: We conducted a study to evaluate the effectiveness of Open AI's GPT models in assessing text summaries generated by six transformer-based models from Hugging Face. The models we analyzed were Distilbart, BERT, Prophetnet, T5, BART, and PEGASUS, and we focused on important qualities of intelligent reporting such as conciseness, relevance, coherence, and readability. We used established evaluation metrics like ROUGE and Latent Semantic Analysis (LSA) to quantitatively assess the generated summaries. Additionally, we employed GPT in a unique way, not as a summarizer but as an evaluator, allowing it to evaluate the summaries independently without explicit metrics. Our findings revealed a strong correlation between GPT evaluations and the traditional metrics, with relevance and coherence evaluation being particularly noteworthy. In conclusion, GPT shows promise as an evaluator and complements traditional evaluation metrics in assessing the quality of text summaries. Our study provides valuable insights for future research in natural language processing and aids in comparing the effectiveness of different transformer-based models in generating high-quality summaries.

Keywords: keyword 1; keyword 2; keyword 3 (List three to ten pertinent keywords specific to the article; yet reasonably common within the subject discipline.)

2. Introduction

In today's digital age, we swim in an ocean of incredibly vast and complex information. This provides a unique challenge for the intelligence community, as the success or failure of attaining strategic goals may depend on how well and swiftly this information is processed and summarized. Creating personalized daily reports that gather various data and present it in a clear, organized, and useful summary is essential. With the growing demand for such systems that can automate text summarizing on a large scale without sacrificing quality or relevance, they are becoming more important in this context. The task of text summarization is becoming increasingly important in natural language processing (NLP), as it has various applications in different fields such as news aggregation and providing condensed versions of lengthy documents [1]. As data continues to grow exponentially, text summarization can greatly improve the accessibility and understanding of content, enabling users to navigate vast amounts of information more efficiently [2].

In recent years, transformer models have become increasingly popular for text summarization tasks. This is due to their ability to capture complex relationships within text data, as demonstrated by Vaswani et al. [3]. These models use self-attention mechanisms and have shown excellent performance in various NLP tasks, marking a significant departure from conventional sequence-to-sequence models [3]. However, there is a significant variation among transformer models, each with unique characteristics and execution traits.

Citation: Mardini, MT.; Lastname, F.; Lastname, F. Title. *Journal Not Specified* **2023**, *1*, 0. https://doi.org/

Received:
Revised:
Accepted:
Published:

Submitted to *Journal Not Specified* for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

Therefore, this study aims to compare the performance of different transformer-based models for text summarization using a well-established dataset.

The aim of this study is to assess the effectiveness of Open AI's GPT models in evaluating text summaries from six transformer-based models provided by Hugging Face[1]. These models include Distilbart [4][2], BERT (Bidirectional Encoder Representations from Transformers) [5], Prophetnet [6], T5 (Text-to-Text Transfer Transformer) [7], BART (Bidirectional and Auto-Regressive Transformers) [4], and PEGASUS (Pre-training with Extracted Gap-sentences for Abstractive Summarization Sequence-to-sequence) [8]. We will evaluate these models based on fundamental properties of intelligence reporting, such as conciseness, relevance, coherence, and readability. To provide a quantitative assessment of the summaries generated, we will use established evaluation metrics such as ROUGE and Latent Semantic Analysis (LSA). Additionally, we will use GPT in a novel way, not as a summarizer but as an evaluator, to evaluate the summaries on its own without explicit metrics. Our main goal is to compare the quality of the summaries from these models.

Our research aims to bridge the existing gap between traditional summary evaluation techniques and the latest advancements in AI technologies. Our goal is to explore how these technologies can enhance the capabilities of the intelligence community while also maximizing the creation of personalized daily reports in an increasingly data-driven society.

3. Background and Literature Review

Text summarization is the process of converting a longer text into a concise version while retaining its essential information, which continues to be a pivotal research area in NLP [9]. It finds broad applicability across domains such as news reporting, automated report generation, and conversation analysis [1]. Earlier, text summarization techniques were rule-based and involved heuristics such as extracting sentences that contain the most frequent terms [10]. Nonetheless, these techniques could not satisfactorily capture the complexity and semantics of the natural language, provoking the development of machine learning-based approaches [11].

Recurrent Neural Networks (RNNs), such as Long Short-Term Memory (LSTM) units, were among the first machine learning methods applied for summarization tasks due to their ability to capture temporal dependencies in text [12]. However, these models often faced challenges with long sequences due to vanishing and exploding gradient problems. Nallapati et al. [13] proposed a novel extractive summarization model named SummaRuNNer, which leveraged a hierarchical RNN structure to generate summaries. The advent of transformer models marked a significant advancement in the field of NLP [3]. They were able to capture contextual information more effectively than their RNN counterparts due to their distinctive architecture, which was centered on the self-attention mechanism. BERT, a transformer model proposed by Devlin et al. [5], revolutionized the way text representation was learned by training on a large corpus of text in a self-supervised manner. However, BERT was designed to generate embeddings for downstream tasks, not to generate text.

Building upon the transformer architecture, several models were proposed that could both understand and generate text, making them suitable for summarization tasks. Notably, the BART model by Lewis et al [4] and the T5 model by Raffel et al. [7] were shown to perform exceptionally well on summarization benchmarks. Other models like PEGASUS Zhang et al. [8] and ProphetNet Yan et al. [6] introduced additional pretraining objectives that proved beneficial for summarization.

Furthermore, the application of knowledge distillation for creating efficient and performant models like DistilBART has paved the way for the practical implementation of large transformer models in resource-constrained environments [14].

[1] https://huggingface.co/
[2] https://huggingface.co/sshleifer/distilbart-cnn-12-6

This study contributes to the ongoing discourse by providing a comparative analysis of various transformer models on a summarization task using the CNN/Daily Mail dataset[15].

4. Methodology

In this study, six transformer-based models, shown in Figure 1 for text summarization were evaluated using two methods. The first method, called ChatGPT-Based evaluation, involved utilizing Open AI's GPT models, while the second method, called metrics-based evaluation, employed evaluation metrics commonly used in the field of NLP. The rational of including the metrics-based method is provide a comparative benchmark to the ChatGPT-based evaluation.

The utilized large language models possess unique architectures and pre-training strategies that have proven beneficial in various NLP tasks. They offer distinctive insights and approaches, setting themselves apart through innovative techniques. To conduct this study, the CNN/daily mail dataset was used. This dataset is widely used as a benchmark in the text summarization field [16] and contains a vast collection of online news articles.

DistilBART (sshleifer/distilbart-cnn-12-6) [3] DistilBART is a smaller, more efficient version of the BART model. It uses a technique known as knowledge distillation, which was first introduced by Hinton et al., where the distilled model (DistilBART) is trained to predict the output of the original model (BART) in an attempt to retain its performance while being more resource-efficient. This version of DistilBART has been fine-tuned on the CNN/DailyMail dataset for the specific task of summarization Lewis et al. [4] and Hinton et al. [17]

BERT-small2BERT-small (mrm8488/bert-small2bert-small-finetuned-cnn_daily_mail-summarization) [4] This model utilizes a smaller version of BERT, a transformer-based model that introduced the idea of bi-directional training in transformers. BERT has been a milestone in transformer-based models due to its impressive performance in various NLP tasks Devlin et al. [5]

ProphetNet (microsoft/prophetnet-large-uncased-cnndm) [5] ProphetNet introduces a novel self-supervised objective called future n-gram prediction and an n-stream self-attention mechanism. These additions allow ProphetNet to predict more than one token ahead during pre-training, which has been shown to be beneficial for downstream sequence generation tasks Qi et al. [6]

T5-small (t5-small) [6] T5, or Text-to-Text Transfer Transformer, reframes all NLP tasks into a text-to-text format, making the model's application flexible across a variety of tasks. The 'small' variant is a more lightweight version, maintaining good performance with fewer parameters Raffel et al. [7]

BART-large (facebook/bart-large-cnn) [7] BART is a transformer-based model that uses a denoising autoencoder for pre-training. Unlike other models that pre-train on a specific task, BART learns to reconstruct the original text by transforming it, leading to an impressive performance on many downstream tasks Lewis et al. [4]

PEGASUS (google/pegasus-cnn_dailymail) [8] PEGASUS, or Pre-training with Extracted Gap-sentences for Abstractive Summarization, utilizes a novel pre-training objective where certain sentences are removed and the model is tasked with generating them. This approach has proved highly effective for abstractive text summarization tasks Zhang et al. [8]

[3] https://huggingface.co/sshleifer/distilbart-cnn-12-6
[4] https://huggingface.co/mrm8488/bert-small2bert-small-finetuned-cnn_daily_mail-summarization/blob/main/README.md?code=true
[5] https://huggingface.co/microsoft/prophetnet-large-uncased-cnndm
[6] https://huggingface.co/t5-small
[7] https://huggingface.co/facebook/bart-large-cnn
[8] https://huggingface.co/google/pegasus-cnn$_{d}ailymail$

To implement our project, we utilized Hugging Face's Transformers library [18]. This library provides pre-trained models and high-level APIs for text processing tasks. We used the EncoderDecoderModel for the BERT-small2BERT-small model, while the pipeline function was used for the other models. To generate a summary using the BERT-small2BERT-small model, we created a function called generate_summary. This function takes a text as input, tokenizes it using BertTokenizerFast, and generates a summary using the EncoderDecoderModel. We tokenized the text with a maximum length of 512 tokens, applying padding and truncation when necessary. The generated summaries were decoded by removing the special tokens. For the remaining models, we utilized Hugging Face's pipeline function to streamline the summarization process. We input each text into the pipeline for its respective model, and the resulting output (a dictionary) was processed to extract the summary text.

The performance of each model was evaluated based on the quality of summaries generated for the first 30 articles from the CNN/Daily Mail dataset that contained fewer than or equal to 512 words. The generated summaries were stored in a Pandas DataFrame Reback et al. [19], with each row representing an article and each column a model's summary.

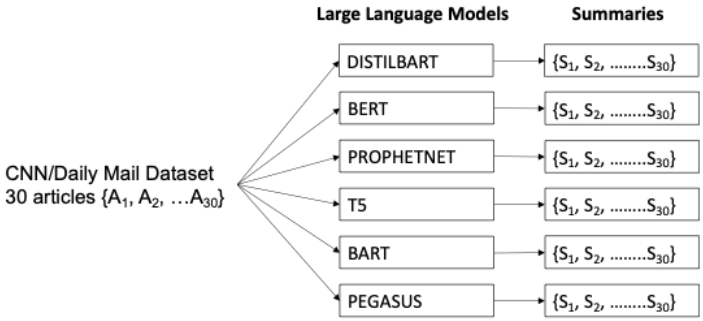

Figure 1. Generation of summaries.

4.1. ChatGPT-Based Evaluation

To evaluate the quality of the generated summaries, we selected four properties including conciseness, relevance, coherence, and readability. Then, we mapped these properties to comparative benchmarks or metrics as shown in Table 1. We utilized the API of Open AI to prompt GPT 3.5 model to evaluate the provided summary as illustrated in Figure 2. Following is an example the prompt that we passed to GPT:

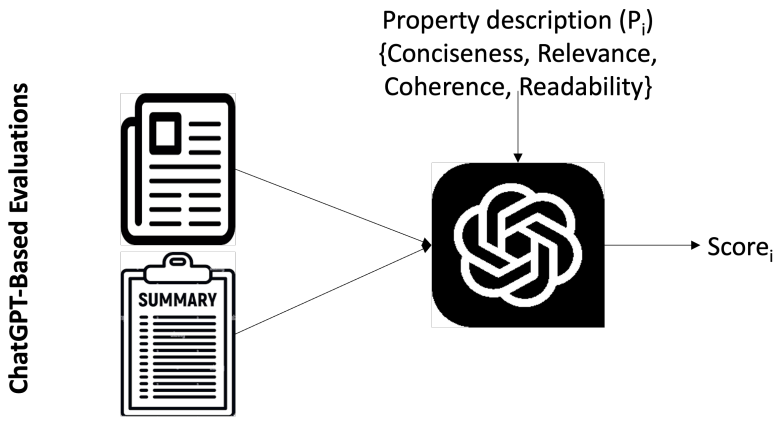

Figure 2. Illustration of the ChatGPT-Based evaluation.

Prompt: Score the following summary given the corresponding article with respect to consistency from 0 to 1 where 1 means most consistent. Note that consistency measures

how much information included in the summary is present in the source article.
Article: [Article]
Summary: [Summary]
Score:

Table 1. Evaluation of the summaries using traditional metrics.

Property	Description	Evaluation metric
Conciseness	A high-quality summary should effectively convey the most important information from the original source while keeping the length brief	Compression Ratio - Calculate the ratio of the length of the summary to the length of the original text.
Relevance	The information presented in the summary should be relevant to the main topic.	ROUGE (Recall-Oriented Understudy for Gisting Evaluation) - compares n-gram overlap (unigrams, bigrams, etc.) between the summary and the reference summaries or the source text. It assesses how well the summary captures important content
Coherence	A good summary should have a clear structure and flow of ideas, making it easy to understand and follow	Latent Semantic Analysis (LSA) to assess the logical connections between sentences or concepts
Readability	The sentence used in the summary should be clear and easily understandable	Flesch-Kincaid to assess the complexity of sentences in the summary.

4.2. Metrics-Based Evaluation

To evaluate summaries based on metrics, we calculated compression ratio, ROUGE (Recall-Oriented Understudy for Gisting Evaluation), Latent Semantic Analysis (LSA), and Flesch-Kincaid for each one of the summaries as detailed in Table 1 and illustrated in Figure 3.

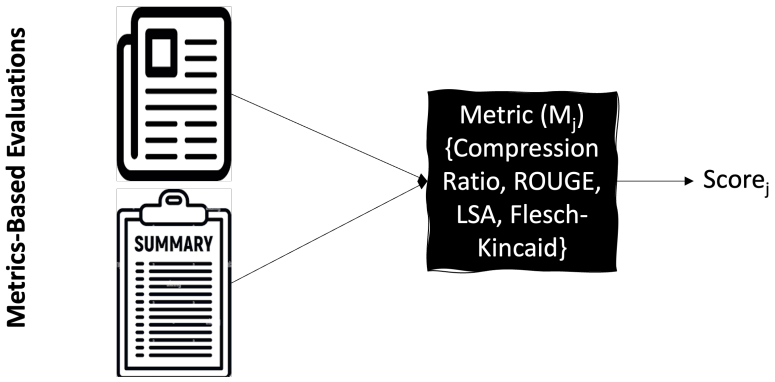

Figure 3. Illustration of the Metrics-Based evaluation.

4.3. Statistical analysis

We compared conventional metrics with ChatGPT-generated scores. To determine the extent of agreement and association, we utilized Pearson's correlation, a widely recognized statistical measure for assessing linear relationships between two variables. Our aim was to determine the level of correlation between the conventional metrics and the

ChatGPT-generated scores by running Pearson's correlation. The results from this assessment provided valuable insights into the consistency and alignment of the assessment provided by the two methods. The correlation coefficient obtained from Pearson's analysis served as a quantitative indicator of the similarity between the two sets of scores. A higher correlation coefficient would indicate a strong positive relationship, suggesting that the scores obtained from the conventional metrics and ChatGPT are in close agreement. Conversely, a lower correlation coefficient would suggest a weaker relationship, possibly indicating differences in the evaluations provided by the two methods.

5. Results

In Table 2, you can observe the metric-based scores calculated for each of the LLMs. These scores are the averages across 30 summaries and have been normalized to range between 0 and 1, with 0 denoting the lowest score and 1 representing the highest score. Notably, for the compression ratio score, we complemented it by subtracting it from 1.

Table 3 displays the computed ChatGPT-based scores for each of the LLMs. These scores represent the averages across 30 summaries and have been scaled to fit within the range of [0-1], where 0 signifies the lowest score and 1 indicates the highest score.

Table 4 shows the results of the Pearson's correlation outcomes including the correlation coefficient and the P value for each one of the evaluation properties.

Figure 4 presents the evaluation results of the summaries from all LLMs using the conventional metrics, while Table 5 displays the evaluation outcomes of the summaries from all LLMs using ChatGPT.

Table 2. Metrics results.

Model	Conciseness (Compression Ratio)	Relevance (ROUGE)	Coherence (LSA)	Readability (Flesch-Kincaid)
DISTILBART	0.19	0.36	0.57	0.45
BERT	0.17	0.25	0.56	0.42
PROPHETNET	0.05	0.08	0.29	0.38
T5	0.15	0.29	0.59	0.43
BART	0.16	0.33	0.57	0.40
PEGASUS	0.13	0.28	0.49	0.38

Table 3. Scoring the properties with ChatGPT3.

Model	Conciseness	Relevance	Coherence	Readability
DISTILBART	0.24	0.78	0.70	0.79
BERT	0.31	0.72	0.64	0.73
PROPHETNET	0.35	0.62	0.38	0.69
T5	0.31	0.73	0.59	0.71
BART	0.26	0.81	0.72	0.82
PEGASUS	0.24	0.79	0.68	0.78

Table 4. Correlation .

Property	Correlation coefficient	P Value
Conciseness	-0.65	0.17
Relevance	0.92	0.01
Coherence	0.85	0.03
Readability	0.11	0.83

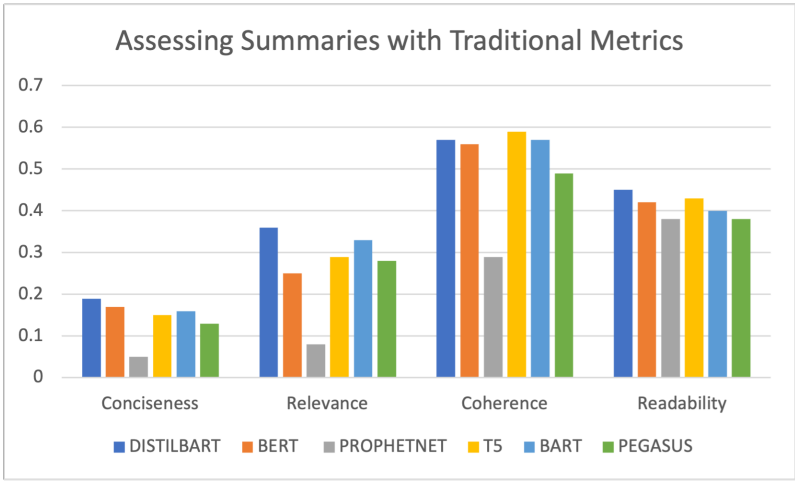

Figure 4. Assessing using traditional metrics.

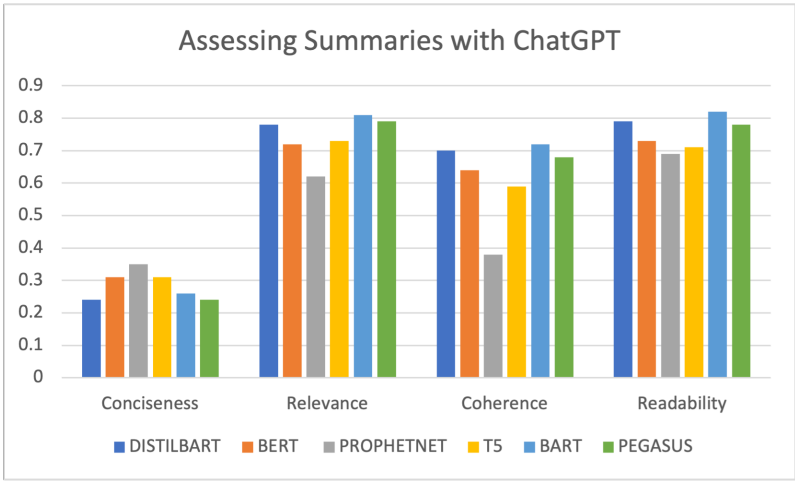

Figure 5. Assessing using ChatGPT.

6. Discussion

In this work, we applied established evaluation metrics such as as ROUGE and Latent Semantic Analysis (LSA). Then, going beyond these conventional approaches, we employed ChatGPT, not as a summarizer but as an evaluator, enabling it to evaluate the summaries on its own without explicit metric direction.

The findings indicate a correlation between the traditional metrics and ChatGPT's evaluations. More specifically, substantial correlation was observed in relation to relevance and coherence, reinforcing the efficacy of ChatGPT in appraising certain elements of summaries.

On comparison, it was observed that ChatGPT generally awards higher scores. This could possibly be due to traditional metrics applying strict criteria in summary evaluation, whereas ChatGPT may take into account additional factors when appraising these summaries.

There's a uniformity in the LLMs when it comes to evaluating summaries. However, the summaries produced using the PROPHETNET model garnered lower scores in both the metrics and ChatGPT evaluations. This was anticipated since these summaries tend to be very concise, occasionally comprised of a single sentence, and hence, fail to provide an adequate representation of the original text.

7. Conclusion

In conclusion, we used Open AI's GPT model in addition to standard metrics like ROUGE and LSA to evaluate summaries. The findings indicate a strong correlation, particularly in terms of relevance and coherence evaluation. GPT tended to give higher scores, likely because it considers other factors. However, PROPHETNET summaries consistently received lower scores due to their brevity and insufficient representation. Overall, GPT has potential as an evaluator and can supplement traditional metrics in evaluating summary quality, providing useful insights for future natural language processing research.

References

1. Liu, Y.; Lapata, M. Text summarization with pretrained encoders. *Association for Computational Linguistics* **2019**.
2. Chouikhi, H.; Alsuhaibani, M. Deep Transformer Language Models for Arabic Text Summarization: A Comparison Study. *Applied Sciences* **2022**, *12*, 11944.
3. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Advances in Neural Information Processing Systems* **2017**, *30*.
4. Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; Zettlemoyer, L. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *Annual Meeting of the Association for Computational Linguistics* **2019**.
5. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *Association for Computational Linguistics* **2018**.
6. Qi, W.; Yan, Y.; Gong, Y.; Liu, D.; Duan, N.; Chen, J.; Zhang, R.; Zhou, M. Prophetnet: Predicting future n-gram for sequence-to-sequence pre-training. *Association for Computational Linguistics* **2020**.
7. Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; Liu, P.J. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research* **2020**, *21*, 5485–5551.
8. Zhang, J.; Zhao, Y.; Saleh, M.; Liu, P. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In Proceedings of the International Conference on Machine Learning. PMLR, 2020, pp. 11328–11339.
9. Widyassari, A.P.; Rustad, S.; Shidik, G.F.; Noersasongko, E.; Syukur, A.; Affandy, A.; et al. Review of automatic text summarization techniques & methods. *Journal of King Saud University-Computer and Information Sciences* **2022**, *34*, 1029–1046.
10. Luhn, H.P. The automatic creation of literature abstracts. *IBM Journal of Research and Development* **1958**, *2*, 159–165.
11. Yadav, A.K.; Singh, A.; Dhiman, M.; Vineet.; Kaundal, R.; Verma, A.; Yadav, D. Extractive text summarization using deep learning approach. *International Journal of Information Technology* **2022**, *14*, 2407–2415.
12. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural computation* **1997**, *9*, 1735–1780.
13. Nallapati, R.; Zhai, F.; Zhou, B. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In Proceedings of the Proceedings of the AAAI conference on artificial intelligence, 2017, Vol. 31.
14. Sanh, V.; Debut, L.; Chaumond, J.; Wolf, T. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108* **2019**.
15. Chen, D.; Bolton, J.; Manning, C.D. A Thorough Examination of the CNN/Daily Mail Reading Comprehension Task. In Proceedings of the Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Berlin, Germany, 2016; pp. 2358–2367. https://doi.org/10.18653/v1/P16-1223.
16. Hermann, K.M.; Kocisky, T.; Grefenstette, E.; Espeholt, L.; Kay, W.; Suleyman, M.; Blunsom, P. Teaching machines to read and comprehend. *Advances in Neural Information Processing Systems* **2015**, *28*.
17. Hinton, G.; Vinyals, O.; Dean, J. Distilling the knowledge in a neural network. *NIPS 2014 Deep Learning Workshop* **2015**.
18. Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; et al. Transformers: State-of-the-art natural language processing. In Proceedings of the Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations, 2020, pp. 38–45.
19. Reback, J.; McKinney, W.; Jbrockmendel, B.J.; Augspurger, T.; Cloud, P.; et al. pandas-dev/pandas: Pandas 1.0. 3: Zenodo; 2020, 2021.