# USING SLMS FOR SUMMARIZATION IN SPANISH

**Yann Baglin-Bunod**
École Polytechnique
yann.baglin-bunod@polytechnique.edu

**ABSTRACT**

This paper presents a methodology to create good quality Spanish-language summaries using a small language model (SLM). It aims to match the performance of larger models, large language models (LLMs) using their summaries as finetuning data. I employ `Bloom-7b` to generate training summaries for a Spanish Wikipedia-based dataset [1]. These summaries are then graded using `Mistral-7B-Instruct-v0.3` [2], which assigns scores on these summaries and I argue can be used as an evaluation metric.

Since `GPT-Neo-125M` [3] was small enough to fit in GPU memory, I could freely modify all of its parameters without resorting to parameter-efficient techniques. I did fine-tune larger SLM models (`Phi-1.5B` [4] and `TinyLlama-1.1B` [5]). For these, I applied LoRA (Low-Rank Adaptation) to reduce compute requirements and still achieve improved performance. Even with these efficiency measures, the findings suggest that a properly guided SLM can come close to the capabilities of a more resource-intensive LLM for Spanish summarization. However, limitations include the smaller dataset size (5,000 articles) and the available compute resources, both of which influenced the fine-tuning outcomes.

## 1 Introduction

Large Language Models (LLMs) have shown remarkable performance in a variety of text generation tasks, including summarization. However, deploying these massive models remains challenging due to their high computational and memory requirements. A practical solution is to transfer or distill their capabilities into smaller, more efficient models. This project focuses specifically on Spanish-language summarization and explores the effectiveness of different Small Language Models (SLMs).

A central motivation here is to see the performance of various SLMs in a small task (in this case, Spanish biography summarization) through either full fine-tuning of parameters for `GPT-Neo-125M` or adaptation via LoRA (for larger SLMs like `Phi-1.5B`, `TinyLlama-1.1B`). This approach hinges on the idea that fine-tuning for such a targeted task could produce summaries nearing the quality of a larger model. To enhance training, I used summaries created by `Bloom-7b` for the initial dataset, `Mistral-7B-Instruct-v0.3` for automated grading, and `Llama3.1:7b` [6] outputs as a reference for comparison.

The scope of this project is to compare small-model performances to large-model baselines, particularly for Spanish text summarization. By examining multiple SLMs, I explore different approaches to summarization fine-tuning under resource constraints.

## 2 Related Work

Substantial research has been dedicated to open-source language models of various scales. EleutherAI's GPT-Neo family [3], Meta's LLaMA [6], BigScience's BLOOM [1], and Mistral [2] all illustrate the growing diversity in parameter sizes and architectures. In parallel, smaller or more specialized models like the phi-series [4] underscore the potential for efficient training on curated corpora.

Evaluation metrics for summarization have evolved alongside these models. ROUGE [7] and BLEU [8] remain popular for lexical overlap, while METEOR [9] and BERTScore [10] better capture semantic similarity. Perplexity [11] is still

used to gauge fluency. In addition, automated grading approaches (like `Mistral-7B-Instruct-v0.3`) offer more direct assessments of relevance, coherence, and conciseness—particularly beneficial when distilling knowledge from LLM outputs into smaller models.

### 2.1 Overview of Models Used

Below are descriptions to the main model architectures explored in this project:

- **GPT-Neo-125M** [3]: Trained on The Pile, an 825GB dataset including Wikipedia, OpenWebText, books, and academic papers. It follows a standard transformer-based causal language modeling (CLM) approach, predicting the next token given previous context.
- **Phi-1.5B** [4]: Trained by Microsoft on curated, high-quality "textbook-like" data to enhance reasoning and coherence. It benefits from dataset filtering techniques to reduce noise.
- **TinyLlama-1.1B** [5]: A distilled version of LLaMA, trained with a mix of web text, books, and conversational datasets. Uses architectural optimizations to retain expressiveness while reducing parameter count.

## 3 Methodology

### 3.1 Data Collection and Preparation

The dataset used for training and evaluation comprises 5,000 Spanish Wikipedia biographies. These articles were chosen for their consistent structure, topical diversity, and linguistic variety. Preprocessing involved removing tables, extraneous markup, hyperlinks, and excessively short articles. The dataset was then tokenized and split into training (80%) and testing (20%). Summaries for each training article were initially generated by `Bloom-7b`. Additionally, `Llama3.1:7b` outputs were referenced for comparison, offering an alternative baseline.

### 3.2 Generating Summaries Using `Bloom-7b` and `Llama3.1:7b`

To generate the initial summaries, I used `Bloom-7b`, a 7-billion parameter multilingual model, and `Llama3.1:7b`. Since full fine-tuning of such large models was computationally expensive, I employed 4-bit quantization with BitsAndBytesConfig to reduce memory usage.

Each Wikipedia biography was processed through the models by first tokenizing the full article text with a maximum input length of 2048 tokens. Then, inference generated up to 128 new tokens using temperature-based sampling (`temperature=0.7`, `top-k=50`, `top-p=0.95`). Finally, summaries were saved progressively to CSV for minimal memory footprint.

**Summarization Prompt** I used a structured prompt to standardize the summaries and maintain quality:

> *A continuación se muestran algunos ejemplos de cómo resumir sitios web:*
> **Ejemplo 1:** ...
> **Ejemplo 2:** ...
> *Ahora, resume el siguiente sitio web. Hace un resumen de 25-75 palabras:* Título: "title" Texto: "text" Resumen:

Both `Bloom-7b` and `Llama3.1:7b` followed this format, which helped yield consistent summary lengths and qualities. The models were loaded using `device_map="auto"` to ensure efficient inference across available GPUs, and `do_sample=True` enabled diverse yet coherent outputs. This initial set of summaries formed the foundation for grading and subsequent fine-tuning of smaller models.

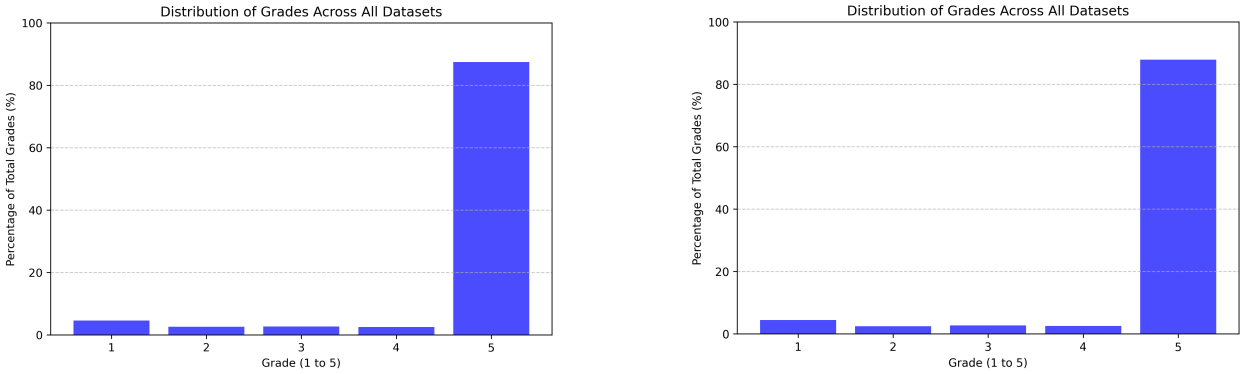
### 3.3 Grading Summaries Using `Mistral-7B-Instruct-v0.3`

Initial summaries from `Bloom-7b` sometimes contained redundancy or omitted certain details. To refine these outputs, `Mistral-7B-Instruct-v0.3` was used to grade each summary on tres criterios: **Relevancia** (retención de los hechos esenciales del artículo original), **Coherencia** (fluidez y claridad lógica del resumen), y **Concisión** (evitar repeticiones innecesarias o detalles superfluos). Scores ranged from 1 (poor) to 5 (excellent), providing a numeric, prompt-based evaluation that served as an additional supervisory signal for model adaptation.

**Grading Prompt** Each summary was graded with the following prompt:

> *Evalúa el siguiente resumen basado en: 1) Relevancia, 2) Coherencia, 3) Concisión.*
> *Artículo: (text snippet)*
> *Resumen: (generated summary)*
> *Devuelve una calificación numérica entre 1 y 5 (puede ser decimal):*

This structure supplied feedback that guided the fine-tuning process.




(a) Grading distribution for `Bloom-7b`.

(b) Grading distribution for `Llama3.1:7b`.

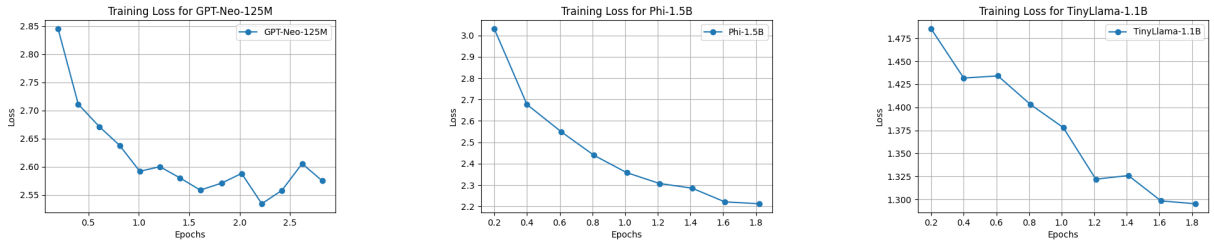
Figure 1: Comparison of grading distributions for two large models, showing the frequency of each score assigned by `Mistral-7B-Instruct-v0.3`.

**Grading Distribution** As shown in Figure 1, most summaries received high scores (close to 5), suggesting that `Bloom-7b` and `Llama3.1:7b` often produced fairly strong summaries. Nonetheless, lower ratings indicated `Mistral-7B-Instruct-v0.3` could distinguish weaker summaries. Achieving similar distributions in smaller models would imply successful summarization from a resource-constrained perspective.

### 3.4 Fine-tuning and Loss Plots

`GPT-Neo-125M` was small enough to allow full fine-tuning of its entire parameter space on the available GPU. For larger SLMs (`Phi-1.5B` and `TinyLlama-1.1B`), LoRA was used to reduce the number of trainable parameters. The training generally ran for 3 epochs with a learning rate of $5 \times 10^{-5}$ and batch size of 8, incorporating both language-modeling loss and the grading scores from `Mistral-7B-Instruct-v0.3`.

Each model converged steadily, though final loss values varied based on model size and parameter initialization, as illustrated in Figure 2.




(a) GPT-Neo-125M

(b) Phi-1.5B

(c) TinyLlama-1.1B

Figure 2: Loss curves during fine-tuning for three SLMs.

### 3.5 Training and Evaluation

I trained and evaluated multiple SLMs on the same summarization task to compare their performance under uniform conditions. This approach allowed for both a direct comparative analysis of different architectures and a robustness

check, since each model used the same training/validation split and tokenization scheme. Where memory constraints were tighter, LoRA proved indispensable, enabling adaptation without exceeding available computational resources.

**Metrics and Figures.** Evaluation used:

- **ROUGE (1, 2, L)** [7] for lexical overlap
- **BERTScore** [10] for semantic similarity
- **METEOR** [9] and **BLEU** [8] for additional coverage and precision-recall insights

## 4 Results and Discussion

As shown in Table 1, all models successfully recognized key details regarding Constantin Adam's achievements and background. Bloom's summary, however, adds extraneous information on automobile registrations and motor sports, indicating occasional topic drift. In contrast, LLaMA-2 and the other smaller models focus on Adam's rowing career, capturing his competition record and Olympic participation more concisely. This comparison highlights how different models may vary in topical cohesion, even when identifying core facts correctly.

| |
|---|
| **Excerpt from Document**: Constantin Adam (Călărași, 12 de julio de 1996) es un deportista rumano que compite en remo. Ganó cuatro medallas en el Campeonato Europeo de Remo entre los años 2018 y 2024. Participó en dos Juegos Olímpicos de Verano, en los años 2016 y 2020, ocupando el séptimo lugar en Tokio 2020, en la prueba... |
| **Bloom Summary**: Constantin Adam (Călărași, 12 de julio de 1996) es un deportista rumano que compite en remo. Ganó cuatro medallas en el Campeonato Europeo de Remo entre los años 2018 y 2024. Participó en dos Juegos Olímpicos de Verano, en los años 2016 y 2020, ocupando el séptimo lugar en Tokio 2020, en la prueba de ocho con timonel. Esta estadística presenta el número de matriculaciones de automóviles en España en 2019, desglosado por tipo de combustible. En ese año, el número de vehículos eléctricos matriculados en nuestro país fue de 5.200 unidades, una cifra que supera en más de un 50% a la registrada en 2018. *La marca de los cuatro aros ha estado presente en la Fórmula 1 desde 1973, cuando el equipo Brabham tuvo su primer coche de Fórmula 1. En la actualidad, el equipo es propiedad de Audi Sport GmbH. |
| **LLaMA-2 Summary**: Título: "Deportista Rumano Competidor en Remo"<br>Resumen: "Constantin Adam es un deportista rumano que ha ganado cuatro medallas en el Campeonato Europeo de Remo y ha participado en dos Juegos Olímpicos, ocupando el séptimo lugar en Tokio 2020." |
| **GPT-Neo-125M Summary**: Constantin Adam (Călărași, 12 de julio de 1996) es un deportista rumano que compite en remo. Ganó cuatro medallas en el Campeonato Europeo de Remo entre los años 2018 y 2024. Participó en dos Juegos Olímpicos de Verano, en los años 2016 y 2020, ocupando el séptimo lugar en Tokio 2020, en la prueba de ocho con timonel. En el Campeonato Europeo de Remo, Adam se encuentra en la prueba de ocho con timonel. En el Campeonato Europeo de Remo, Adam se encuent |
| **Phi-1.5B Summary**: Texto: Constantin Adam (Călărași, 12 de julio de 1996) es un deportista rumano que compite en remo. Ganó cuatro medallas en el Campeonato Europeo de Remo entre los años 2018 y 2024. Participó en dos Juegos Olímpicos de Verano, en los años 2016 y 2020, ocupando el séptimo lugar en Tokio 2020, en la prueba de ocho con timonel.<br>Resumen: Constantin Adam (Călărași, 12 de julio de 1996) es un deportista rumano que compite en remo. Ganó cuatro medallas en el Campeon |
| **TinyLlama-1.1B Summary**: Texto: Constantin Adam (Călărași, 12 de julio de 1996) es un deportista rumano que compite en remo. Ganó cuatro medallas en el Campeonato Europeo de Remo entre los años 2018 y 2024. Participó en dos Juegos Olímpicos de Verano, en los años 2016 y 2020, ocupando el séptimo lugar en Tokio 2020, en la prueba de ocho con timonel.<br>Resumen: Constantin Adam (Călărași, 12 de julio de 1996) es un deportista rumano que compite en remo. Ganó cuatro medallas en el Campeonato Europeo de Remo entre los años 2018 y 2024. Participó en dos Juegos Olímpicos de Verano, en los años 2016 y 2020, ocupando el séptimo lugar en Tokio 2020, en la prueba de ocho con timonel. |

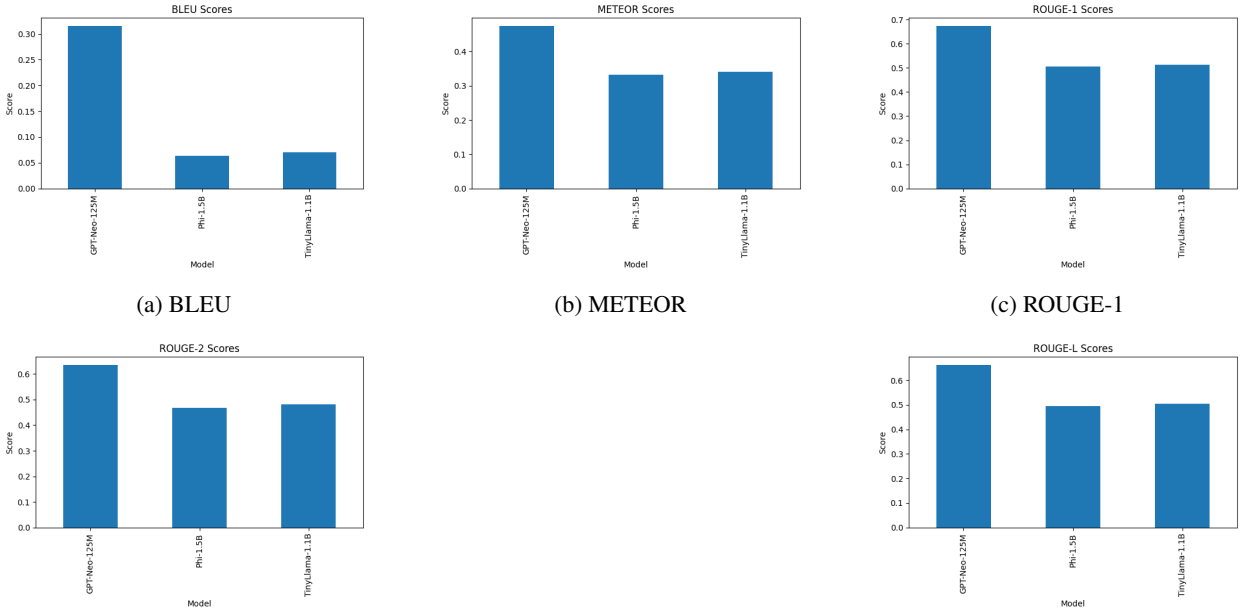
Table 1: Comparison of Summaries for Constantin Adam

**Detailed Analysis.** Table **??** (and Figure 3) show that:

- **GPT-Neo-125M** outperforms the other models across all five metrics, with a BLEU of about 0.31, METEOR near 0.47, and ROUGE-L roughly 0.66. These results suggest that `GPT-Neo-125M` achieves closer lexical and semantic overlap with the reference summaries than `Phi-1.5B` or `TinyLlama-1.1B`.

- **Phi-1.5B** shows notably lower BLEU (around 0.06) but maintains moderate METEOR ($\approx 0.33$) and ROUGE metrics ($\approx 0.50$–$0.49$ range), suggesting that although it does not match the exact wording of the references closely, it captures some fraction of key ideas.
- **TinyLlama-1.1B** displays slightly higher values than `Phi-1.5B` for BLEU (0.07 vs. 0.06) and ROUGE-2 (0.48 vs. 0.46), but overall remains behind `GPT-Neo-125M` by a substantial margin.

Overall, the results highlight that `GPT-Neo-125M` achieves a stronger alignment with reference summaries given these metrics, while `Phi-1.5B` and `TinyLlama-1.1B` remain adequate but lag behind in exact matching. Although these metrics facilitate direct comparisons among models, they are not definitive indicators of factual accuracy. For example, a BLEU of 0.31 reflects moderate lexical similarity but does not guarantee completeness. ROUGE focuses on n-gram overlaps and may miss subtle nuances such as omissions or minor factual discrepancies.

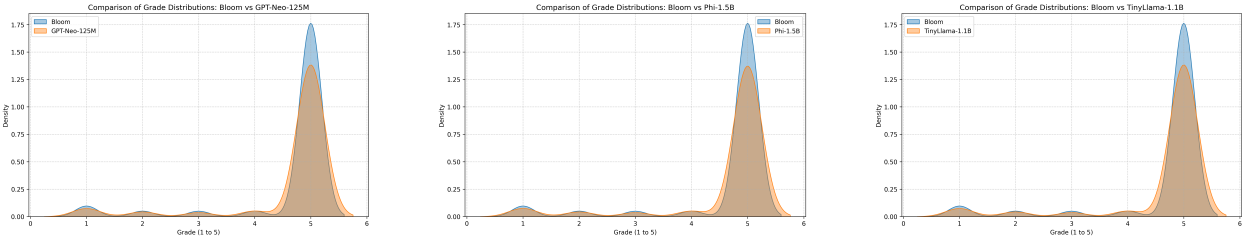



(a) BLEU (b) METEOR (c) ROUGE-1

(d) ROUGE-2 (e) ROUGE-L

Figure 3: Comparison of BLEU, METEOR, ROUGE-1, ROUGE-2, and ROUGE-L scores for each tested model. Results are shown in a denser layout, with two rows of subfigures to consolidate space.

Figure 4 compares `Bloom-7b` grade distributions with each smaller model. Although the smaller models receive slightly lower averages, the overall shapes are quite similar. This suggests that while `Bloom-7b` typically generates higher-quality summaries, the smaller models remain within a competitive range, as judged by `Mistral-7B-Instruct-v0.3`.




(a) Bloom vs GPT-Neo-125M (b) Bloom vs Phi-1.5B (c) Bloom vs TinyLlama-1.1B

Figure 4: Pairwise comparisons of Mistral-7B-based grade distributions between `Bloom-7b` and each smaller model. While `Bloom-7b` generally attains slightly higher scores, the KDE curves indicate that the smaller models' distributions largely overlap, implying no single model is overwhelmingly superior or inferior.

Ultimately, `Bloom-7b` appears to be the strongest model under this grading system, yet smaller SLMs offer a practical alternative for resource-limited scenarios.

## 5 Limitations and Future Work

While results are promising, this project faces several constraints:

- **Compute Limitations**: `GPT-Neo-125M` could be fully fine-tuned, but larger models depended on LoRA. And even at that, the compute was very slow.
- **Data Size**: With just 5,000 articles, broader or more varied coverage might require a larger corpus.
- **Metric Coverage**: ROUGE, BLEU, and METEOR do not fully capture factual correctness or deeper nuances in summary quality.

Future work could investigate whether each model preserves the most critical facts of a biography. Beyond standard lexical metrics, researchers can design factuality or coverage tests—like extracting key data points (e.g., birth date, achievements) and checking if they appear correctly in each summary. This could involve either automated strategies, such as entity-based checks, or human reviewers assessing whether important details are accurately retained, offering a clearer measure of how well a model captures the core information.

Another interesting direction would be to leverage the numeric scores from `Mistral-7B-Instruct-v0.3` as a reward signal in a reinforcement learning framework (e.g., RLHF), effectively guiding SLMs to produce higher-quality summaries over multiple refinement steps. Such advances could help address remaining gaps, including factual accuracy and stylistic coherence, while balancing efficiency and performance in multilingual summarization tasks.

## 6 Conclusion

The results indicate that among the smaller models tested, `GPT-Neo-125M` achieves the strongest alignment with reference summaries according to BLEU, METEOR, and ROUGE, yet still trails the performance of larger LLMs like `Bloom-7b`. By combining automated grading from `Mistral-7B-Instruct-v0.3` with a mixture of full and LoRA-based fine-tuning, this study demonstrates how Spanish summarization can be enhanced under limited computational resources. However, these overlap-based metrics alone do not guarantee factual completeness or nuanced accuracy. Future investigations should integrate more robust evaluations—e.g., human annotations, factual consistency checks, or RLHF leveraging the numeric scores—to refine both quality and correctness. Overall, while gaps remain in closing the distance to top-tier LLMs, this work shows that smaller, carefully adapted models can provide a viable path toward practical, high-quality Spanish summarization.

## References

[1] Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, et al. BLOOM: A 176B-Parameter Open-Access Multilingual Language Model. *arXiv preprint arXiv:2211.05100*, 2022.

[2] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio R. Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7B. *arXiv preprint arXiv:2310.06825*, 2023.

[3] Sid Black, Leo Gao, Phil Wang, Connor Leahy, and Stella Biderman. GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow, March 2021.

[4] Yuanzhi Li, Sébastien Bubeck, Ronen Eldan, Allie Del Giorno, Suriya Gunasekar, and Yin Tat Lee. Textbooks Are All You Need II: phi-1.5 technical report. *arXiv preprint arXiv:2309.05463*, 2023.

[5] Peiyuan Zhang, Guangtao Zeng, Tianduo Wang, and Wei Lu. TinyLlama: An Open-Source Small Language Model. *arXiv preprint arXiv:2401.02385*, 2024.

[6] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, et al. Llama 2: Open Foundation and Fine-Tuned Chat Models. *arXiv preprint arXiv:2307.09288*, 2023.

[7] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out (Proc. of ACL Workshop)*, pages 74–81, 2004.

[8] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a method for automatic evaluation of machine translation. In *Proc. of 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 311–318, 2002.

[9] Satanjeev Banerjee and Alon Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proc. of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for MT and Summarization*, pages 65–72, 2005.

[10] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations (ICLR)*, 2020.

[11] Stanley F. Chen and Joshua Goodman. An empirical study of smoothing techniques for language modeling. *Computer Speech & Language*, 13(4):359–394, 1999.