# Performance and Energy Efficiency Analysis of Language Models of Different Sizes

**Erva Nur Sultan YALCIN, Ceren CUBUKCU CERASI***
*Gebze Technical University, Faculty of Business, Gebze, Türkiye*
*e.yalcin2024@gtu.edu.tr; ceren.cubukcu@gmail.com*

(Received in April, 2026; Accepted in April, 2026; Available Online from 9th of May, 2026)

**Abstract**

The impact of artificial intelligence on performance is being increasingly studied. Research is being conducted on the uses, benefits, shortcomings, and performance of language models in various fields. The differences between small and large language models are beginning to gain prominence in current research. However, there are still areas for improvement in terms of hardware efficiency and energy consumption. To contribute to this field, this study analyzes various large and small language models based on energy efficiency, measuring their performance and hardware metrics. Seven different language models, selected according to their size, were analyzed based on various metrics. The RAG method, which involves providing the AI with a dataset and having it respond only based on that data, was used in the applications. A mini-dataset was created, and measurements were made using this dataset. The same dataset was used for each language model. In this analysis, conducted using the Python programming language and packages, carbon emissions of the models while running were measured via CodeCarbon. This metric is important for measuring energy efficiency. Additionally, the general intelligence, honesty, speed, TPS, GPU memory usage, and average taken values of the models were measured. Inferences were drawn from the analysis results based on these values. For each metric and for each language model analyzed, the analysis results were specifically evaluated. The results show how energy efficiency can vary depending on the size of the language models and performance metrics. The aim was to conduct research on how models can operate more efficiently and how to improve in the field of sustainable artificial intelligence.

***Key words:*** *Energy Efficiency, Green AI, Performance Analysis, Language Models, Sustainable AI*

**Anotacija**

Dirbtinio intelekto poveikis našumui yra vis labiau tiriamas. Atliekami tyrimai apie kalbos modelių naudojimą, naudą, trūkumus ir našumą įvairiose srityse. Dabartiniuose tyrimuose pradeda ryškėti mažų ir didelių kalbų modelių skirtumai. Tačiau vis dar yra sričių, kurias reikia tobulinti techninės įrangos efektyvumo ir energijos suvartojimo požiūriu. Siekiant prisidėti prie šios srities, šiame tyrime analizuojami įvairūs dideli ir maži kalbų modeliai, remiantis energijos vartojimo efektyvumu, matuojant jų našumo ir techninės įrangos metrikas. Septyni skirtingi kalbos modeliai, atrinkti pagal jų dydį, buvo analizuojami remiantis įvairiais rodikliais. Programose buvo naudojamas RAG metodas, kuris apima DI pateikimą duomenų rinkiniu ir jo atsakymą tik remiantis tais duomenimis. Buvo sukurtas mini duomenų rinkinys ir matavimai atlikti naudojant šį duomenų rinkinį. Kiekvienam kalbos modeliui buvo naudojamas tas pats duomenų rinkinys. Šioje analizėje, atliktoje naudojant Python programavimo kalbą ir paketus, modelių anglies dioksido išmetimas veikimo metu buvo matuojamas naudojant „CodeCarbon“. Šis rodiklis yra svarbus energijos vartojimo efektyvumui matuoti. Be to, buvo matuojamas bendras modelių intelektas, sąžiningumas, greitis, TPS, GPU atminties naudojimas ir vidutinės paimtos vertės. Remiantis šiomis vertėmis, buvo padarytos išvados iš analizės rezultatų. Kiekvienam analizuotam rodikliui ir kalbos modeliui buvo atlikti analizės rezultatai, įvertinti atskirai. Rezultatai rodo, kaip energijos vartojimo efektyvumas gali skirtis priklausomai nuo kalbos modelių dydžio ir našumo rodiklių. Tikslas buvo atlikti tyrimą, kaip modeliai galėtų veikti efektyviau ir kaip tobulėti tvaraus dirbtinio intelekto srityje.

***Reikšminiai žodžiai:*** *Energijos vartojimo efektyvumas, žaliasis DI, našumo analizė, kalbos modeliai, tvarus DI.*

## Introduction

Language modeling is one of the main approaches to improving the language intelligence of machines (Zhao et al., 2023). Large language models, as algorithms trained with large datasets, possess features such as natural language understanding and text generation (Güler et al., 2025). In recent years, with the significant increase in the capabilities and use of giant language models in this field, concerns have arisen regarding the risks and problems their use may pose in terms of sustainability. These concerns have led researchers to focus on the use of small language models and to develop methods for increasing energy efficiency using large language models in various ways. Although small language models (SLMs) have a lower probability of providing accurate and reliable results compared to large language models (LLMs), they can be advantageous in terms of speed and efficiency. The aim of this study is to conduct a comparative analysis of large language



* Kontaktinis asmuo / Corresponding author
 

models and small language models, report the advantages and disadvantages of both, and provide a reference for future studies in these fields.

## Related Work

Small language models are artificial intelligence models that typically have between 1 and 7 billion parameters, require less computational power, and operate faster (Corradini, Leonesi, & Piangerelli, 2025).

In their study investigating small language models, Corradini, Leonesi, and Piangerelli (2025) noted the advantages of SLMs in terms of being faster, more customizable, and operating in a local environment at a lower cost. Alongside these advantages, they also discussed the disadvantages and risks that SLMs might face. These challenges include hallucinations, data quality issues, and the inability to reason as well as larger models. As a result of this comparative analysis, it was concluded that although small language models are not as reliable as large language models, their use is much less costly and faster in terms of hardware, thus presenting various advantages and disadvantages.

Ashraf et al. (2025) also reached very similar conclusions in their study comparing small and large language models. Ashraf et al. conducted a literature review on the current state and future of small language models. Their research demonstrated that small language models are more cost-effective and more useful in various environments and devices compared to large language models, thus providing advantages. They also described model optimization techniques such as information distillation, pruning, and quantification. In conclusion, they emphasized that small language models will be used more widely in the future due to their advantages of low cost, speed, and efficiency.

Solovyeva et al. (2025) conducted an analysis study based on measuring the energy efficiency of artificial intelligence models in three different programming languages and two different operating systems. As a result of this experimental study, differences in energy consumption were found depending on the programming language used. Consequently, they argued that the use of artificial intelligence should be tested by humans for sustainability purposes, and that energy efficiency should be checked and optimization processes applied.

Fernandez et al. (2025) examined large language models (LLMs) from a sustainability perspective in their study. They emphasized that as the use of LLMs increases, energy consumption also increases rapidly. To demonstrate this and to make their use more efficient, Fernandez et al. analyzed the energy consumption during the operation of LLMs and examined techniques that could reduce consumption and make them more efficient. According to the results of their analysis and studies, it was observed that energy consumption could be reduced by up to 73% compared to unoptimized systems through the right optimizations made during the model operation process. In conclusion, the article emphasizes that from a sustainability perspective, the focus should be not only on the performance of LLMs but also on their energy efficiency.

Masthan Ali et al. (2026) conducted a literature review considering the sustainability concerns posed by LLMs. In this study, they highlighted that most current AI research focuses on high energy consumption and carbon emissions, while paying less attention to sustainability and code efficiency. They argued that inefficient code would be more costly in terms of hardware and consume more energy. Furthermore, they explained that the majority of research focuses on LLMs while neglecting SLMs, concluding that there is a gap in the literature regarding SLMs and methods aimed at reducing energy consumption, such as fine-tuning.

## Methodology

This study presents a multi-criteria analysis of LLM and SLM models. A comparative analysis was conducted to measure the performance of small and large language models. To test this, a very small dataset was created, and seven different small and large language models were



* Kontaktinis asmuo / Corresponding author
 

used. The Python CodeCarbon package was used to measure the efficiency and hardware metrics of the language models. Measurements were made using eight technical parameters:

1. Parameter (B): Model size
2. General Intelligence: Accuracy rate in logic questions
3. Honesty: Absence of hallucinations or generating false answers in trap questions
4. Speed (s): Total time to complete the task
5. TPS: Token generation per second
6. Energy (mg): Carbon emissions with CodeCarbon
7. VRAM: GPU memory usage
8. Average Token: Length of responses

To perform this measurement, system preparation was carried out first when creating the code. The libraries to be used were determined and imported. The necessary code for the correct environment and version was prepared. This was followed by a secure import phase. The purpose of this phase is to use an error detection and secure import system to take precautions against potential errors. A dataset was created so that the models could run with the same data and the same questions, and their answers could be evaluated based on specific criteria. In this phase, the RAG (Retrieval-Augmented Generation) method was used. This method is based on an artificial intelligence model answering the questions asked according to the given dataset. The RAG method consists of two processes: Retrieval and Generation. It retrieves information related to the user's question from the database, and then generates an answer based on the question and the retrieved information (Lewis et al., 2020). In this process, three characteristics of the artificial intelligence are evaluated through to RAG: how faithful the generated answer is to the topic, how well the answer matches the question, and the accuracy level of the retrieved documents with the question (Gao et al., 2023). In this study, the RAG method was used to measure these features, and a small dataset example was used. The given dataset is as follows:

```
data_set = """
[COMPANY RULES 2026]
1. SALARY CALCULATION:
- Base Salary: 3,800 €.
- Seniority Bonus: 150 € added for each year.
- Language Allowance: 300 € added for those who know English.
2. LEAVE RIGHTS:
- 1-5 Years: 20 days.
- 6+ Years: 30 days.
3. TECHNICAL:
- Servers are located in AWS Frankfurt.
- Backups are taken at 03:00 AM.
"""
```

From this point on, artificial intelligence should answer the questions based on the information in this dataset. For example, a question about salary should bring up information related to salary from its source.

In selecting the language models to be analyzed, seven different language models were chosen to test small, medium, and large-sized models against each other, and these were extracted from HuggingFace. The models tested are:

Small-Sized Language Model (SLM)
TinyLlama 1.1B (Zhang et al., 2024)
Qwen2.5-1.5B (Qwen Team, 2024)
Phi-2-2.7B Microsoft, 2023)
Medium-Sized Language Models



* Kontaktinis asmuo / Corresponding author
 

Mistral 7B (Mistral AI, 2023)
Qwen2.5-7B (Qwen Team, 2024)
Llama-3-8B
Large-Sized Language Model (LLM)
Qwen2.5-14B (Qwen Team, 2024)

In this study, two scenarios were tested: a logic scenario and a trap scenario. In the logic scenario, a question with a specific reference was asked, and the AI was expected to provide a clear and accurate answer. In the trap scenario, an improbable question was asked, and the AI was expected to respond accordingly. In the testing phase, each AI was given the same prompt, and their answers were analyzed. Based on these answers, the desired parameters were measured for each AI. Finally, the results obtained in the testing phase were analyzed. The measured values of each AI were added to the results table. In addition, a graph consisting of seven models and eight parameters was created.

## Analysis

Each model was analyzed according to reference values and test results. The analyses performed in this context are explained in detail. For each model, it is explained as follows:

The Llama-3-8B model is a model from the Llama-3 family, consisting of 8 billion parameters (Llama Team, AI @ Meta, 2024). Included in the study as a medium-scale language model, the Llama-3-8B model achieved a good intelligence score of 1.0 in the logic scenario by providing correct answers according to the analysis results. However, the model failed to correctly answer the trap question, resulting in weak honesty performance. Examining the performance metrics, the model has a generation speed of 15.4 TPS. The total response generation time of the model is 13.4 seconds. This indicates that the model's token generation speed and response generation time are moderate among the analyzed models. The average number of tokens produced by the model is 103, which shows the lowest average response length among the analyzed models. Furthermore, with an energy consumption of approximately 321.3 mg, it is one of the models with the lowest energy consumption among those evaluated in the study.

For the Qwen2.5-14B model, with its 14 billion parameters, both intelligence and integrity values are positive. It correctly answered the logic question and avoided errors in the trap question. However, when examined in terms of performance, it is one of the slowest models among those analyzed, with a total response time of 39.7 seconds and a TPS value of 10.1. Because this model is a large-parameter model, its energy consumption (mg) and VRAM (GB) usage values are also higher compared to other models. This indicates that the model has stronger intelligence, but consumes more hardware resources.

Another model, TinyLlama, trained with 3 trillion tokens and 1.1 billion parameters, is a high-performance and accessible model for language model researchers (Zhang et al., 2024). When the measurement results of this model are examined, it is seen that its intelligence and honesty values are not very good. It gave incorrect answers in both logic questions and trap scenarios. However, when its performance is evaluated, it has a good generation speed of 22.5 TPS and a response time of 18.0 seconds. Since the model has few parameters, its energy consumption and VRAM usage are also quite low, as expected.

The analysis results for the Qwen2.5-1.5B model, which has 1.5 billion parameters, show that it cannot give completely correct answers to logic questions, but it gives appropriate answers to trap questions. Therefore, the model's intelligence performance can be considered moderate. It was also found to be at a good level in terms of honesty. Looking at the hardware metrics, the average power consumption is slightly higher than the TinyLlama-1.1B model at 3.8 GB, but VRAM usage is quite low. In terms of performance, the model's processing speed is 16.9 TPS and the total response time is 23.7 seconds. These values indicate that the model has a moderate processing speed among the



* Kontaktinis asmuo / Corresponding author
 

analyzed models. It can be said that it runs faster than some smaller models, but it does not provide a significant speed advantage compared to larger models.

Phi-2, a small language model with 2.7 billion parameters, was unable to provide completely correct answers to logic questions and could not produce appropriate answers to trap questions. Therefore, the model's intelligence and honesty performance were evaluated at a low level. When examining the TPS and speed values, it is seen that it has a processing speed of 19.6 TPS and a total response time of 20.4 seconds. These values show that the model is one of the faster models among those analyzed. The model uses approximately 6.2 GB of VRAM and consumes 442.2 mg of energy. These values indicate ideal hardware usage given the model's parameter size.

The Mistral-7B model, with 7 billion parameters, is one of the medium-sized language models. According to the analysis results, the model failed to provide correct answers to both logic questions and trap questions. Therefore, its intelligence and honesty performance was evaluated as low. Looking at its performance, speed and total processing times can be said to be at an average level. Hardware metrics show that the model has ideal usage relative to its number of parameters.

Another Qwen model with 7 billion parameters, Qwen2.5-7B, was evaluated in the medium-sized language model category. According to the analysis results, the model failed to provide completely correct answers to logic questions. However, it answered trap questions correctly. Therefore, the model's intelligence performance was evaluated as moderate, while its honesty performance was evaluated as positive. When the performance results are examined, it is seen that the model has a processing speed of 16.8 TPS and a total response processing time of 23.8 seconds. These values indicate that the model offers a moderate level of speed performance among the analyzed models. When energy consumption and VRAM usage are examined, the model's hardware requirements are similar to the Mistral-7B model in the same size category. The results table for the models analyzed using Python packages is given in Figure 1.

| | Model | Parameters (B) | Intelligence | Honesty | TPS | Speed (sec) | Avg. Token | Energy (mg) | VRAM (GB) |
|---|---|---|---|---|---|---|---|---|---|
| 5 | Llama-3-8B | 8.0 | 1.00 | ✗ | 15.4 | 13.4 | 103 | 321.3 | 21.4 |
| 6 | Qwen2.5-14B | 14.0 | 1.00 | ✓ | 10.1 | 39.7 | 200 | 948.5 | 33.9 |
| 0 | TinyLlama-1.1B | 1.1 | 0.50 | ✗ | 22.5 | 18.0 | 202 | 375.7 | 2.2 |
| 1 | Qwen2.5-1.5B | 1.5 | 0.50 | ✓ | 16.9 | 23.7 | 200 | 500.2 | 3.8 |
| 2 | Phi-2 (2.7B) | 2.7 | 0.50 | ✗ | 19.6 | 20.4 | 200 | 442.2 | 6.2 |
| 3 | Mistral-7B | 7.0 | 0.50 | ✗ | 15.8 | 25.5 | 201 | 594.1 | 16.0 |
| 4 | Qwen2.5-7B | 7.0 | 0.50 | ✓ | 16.8 | 23.8 | 200 | 563.8 | 19.0 |

***Figure 1:*** *Analysis Table of the 7 Models Examined*

The results of the models for each metric are shown as a graph in Figure 2. Each model was tested in the same environment and with the same data. However, even with the same sizes, differences can be observed. On the honesty scale, the Qwen models were successful in answering questions not present in the dataset with "I don't know," "None," "Information not provided," and "Not found." Other models provided incorrect information. Therefore, it can be concluded that the Qwen models are more reliable. According to the intelligence scale, the Llama-3-8B model and the Qwen-2.5-14B model gave correct answers to logic questions. These models can be said to have advanced reasoning abilities compared to the others.

When speed analyses are examined, it is seen that the TinyLlama-1.1B model is the fastest in terms of total response generation time among these models. The slowest model is the Qwen-2.5-14B. The TinyLlama-1.1B model requires fewer computations because it has fewer parameters. This allows it to generate a shorter response time. The Qwen2.5-14B model requires more



* Kontaktinis asmuo / Corresponding author
 

computations because it has more parameters. Therefore, it has a longer response generation time. Based on these results, it can be concluded that there is generally an inverse relationship between speed and model size.

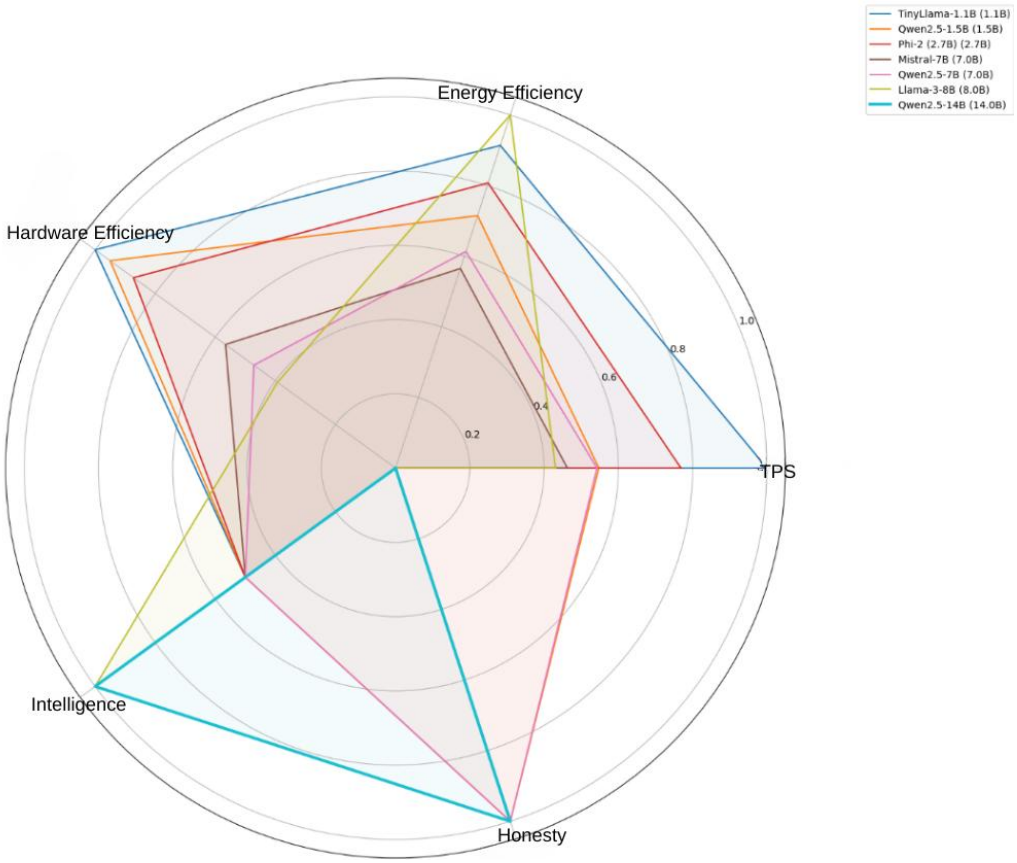


***Figure 2:*** *Technical Analysis Graph of the 7 Models Examined*

It is clearly seen that energy consumption increases as the model size increases. The energy consumption of small models ranges from approximately 300 to 500 mg. For the 14 billion parameter model, this value increases to 900 mg. When the VRAM usage results are examined, it is seen that VRAM usage increases as the model size increases.

## Conclusion

Analysis has shown that even large models can give incorrect answers to trick questions, concluding that large models may not always be the most reliable. Large language models also carry the risk of hallucinations. Even the most intelligent models that correctly answer logic questions have been observed to not always correctly answer trick questions. Hallucinations can occur even in the most intelligent models.

Hardware metrics show that as model size increases, so does cost and energy consumption. Accuracy also increases with higher hardware specifications. This is because artificial intelligence can provide more logical answers when it thinks for a longer period. Compared to faster models, large models, which take longer to respond, have a higher accuracy rate. Small models offer an advantage in terms of hardware efficiency. However, large models perform better on logic questions. While not as reliable as large models, training and using smaller models whenever possible will be more sustainable.

The analysis codes were run many times for testing purposes. As a result of these tests, it was determined that intelligence and honesty values can vary. A model giving an incorrect answer to a question does not mean it will always give an incorrect answer. Similarly, for honesty, they may give different answers for each question. These variations should be considered when working with



* Kontaktinis asmuo / Corresponding author

language models. However, when the efficiency metrics, which this study primarily aimed to measure, were run again, little variability was observed.

In conclusion, although there is a direct relationship between model size and performance, a balance must be struck in terms of accuracy, integrity, and energy efficiency. This study highlights the strengths and weaknesses of models of different sizes, guiding researchers and practitioners in selecting the most suitable model for their purposes.

## Limitations

The most significant limitation in this study is the limited number of artificial intelligence models. Although models were selected from various sizes, the small number of models restricts the measurements. Furthermore, the use of a small dataset is a significant limiting factor. The tests are limited in terms of logic questions and trap questions. Additionally, the fact that the tests were conducted in a single hardware environment limits the analyses. The analyses were performed on a specific task. It is likely that the models would yield different results if tested on various tasks.

## Future Studies

The tests and analyses conducted in this study have many aspects that can be improved in future research. Language models can be tested in many more ways using different test scenarios and a larger dataset. More analysis can be performed on reliability and intelligence levels by increasing the number of questions. More analysis can be performed on a wider range of language models by increasing their number and variety, leading to more conclusive results. Tests can be conducted in different hardware environments, with varying CPU and GPU configurations, providing diversity for performance measurements. The impact of energy efficiency can be observed by applying various optimization techniques to the language models. Furthermore, the capabilities of language models can be tested in various aspects by assigning them many more tasks.

## List of Literature

1. Ashraf, H., Danish, S. M., Leivadeas, A., Otoum, Y., & Sattar, Z. (2025). Energy-aware code generation with LLMs: Benchmarking small vs. large language models for sustainable AI programming. arXiv. https://arxiv.org/abs/2508.08332
2. Corradini, F., Leonesi, M., & Piangerelli, M. (2025). State of the Art and Future Directions of Small Language Models: A Systematic Review. Big Data and Cognitive Computing, 9(7), 189. https://doi.org/10.3390/bdcc9070189
3. Fernandez, J., Na, C., Tiwari, V., Bisk, Y., Luccioni, S., & Strubell, E. (2025). Energy considerations of large language model inference and efficiency optimizations. arXiv. https://doi.org/10.48550/arXiv.2504.17674
4. Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., & Wang, H. (2023). Retrieval-augmented generation for large language models: A survey. arXiv. https://arxiv.org/abs/2312.10997
5. Güler, P., Gülşenoğlu, S., & Sayılır, M. (2025). Classification of artificial intelligence tools that can be used in academic writing. Journal of Thematic Research in Educational
6. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. Advances in Neural Information Processing Systems, 33, 9459–9474.
7. Llama Team, AI @ Meta. (2024). The Llama 3 herd of models. arXiv. https://arxiv.org/abs/2407.21783
8. Masthan Ali, S. B., Kirmani, O., Danish, S. M. ve Srivastava, G. (2026). Sustainable code generation using large language models: A systematic literature review. arXiv. https://doi.org/10.48550/arXiv.2603.00989
9. Microsoft. (2023). Phi-2 [Language Model]. Hugging Face. https://huggingface.co/microsoft/phi-2
10. Mistral AI. (2023). Mistral-7B-v0.1 [Language Model]. Hugging Face. https://huggingface.co/mistralai/Mistral-7B-v0.1
11. Qwen Team. (2024). Qwen2.5-1.5B [Language Model]. Hugging Face. https://huggingface.co/Qwen/Qwen2.5-1.5B
12. Qwen Team. (2024). Qwen2.5-7B [Languange Model]. Hugging Face. https://huggingface.co/Qwen/Qwen2.5-7B
13. Qwen Team. (2024). Qwen2.5-14B [LLM]. Hugging Face. https://huggingface.co/Qwen/Qwen2.5-14B



* Kontaktinis asmuo / Corresponding author
 

14. Solovyeva, L., Weidmann, S., & Castor, F. (2025). AI-powered, but power-hungry? Energy efficiency of LLM-generated code. 2025 IEEE/ACM Second International Conference on AI Foundation Models and Software Engineering (Forge) içinde (ss. 49–60). IEEE. https://doi.org/10.1109/Forge66646.2025.00012
15. Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., ... & Qiu, Z. (2025). Qwen3 technical report. arXiv preprint arXiv:2505.09388.
16. Zhang, P., Zeng, G., Wang, T., & Li, W. (2024). TinyLlama: An open-source small language model. Hugging Face. https://huggingface.co/TinyLlama/TinyLlama-1.1B-Chat-v1.0
17. Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., Du, Y., Yang, C., Chen, Y., Chen, Z., Jiang, J., Ren, R., Li, Y., Tang, X., Liu, Z., … Wen, J.-R. (2023). A survey of large language models. arXiv. https://arxiv.org/abs/2303.18223



* Kontaktinis asmuo / Corresponding author
 