




1 languagemodels: A Python Package for Exploring 2 Modern Natural Language Processing

3 Jonathan L. Craton ¹

4 ¹ Department of Computer Science, Anderson University (IN)

DOI: [10.xxxxxx/draft](https://doi.org/10.xxxxxx/draft)

Software

- [Review](#) 
- [Repository](#) 
- [Archive](#) 

Submitted: 16 May 2023

Published: unpublished

License

Authors of papers retain copyright
and release the work under a
Creative Commons Attribution 4.0
International License ([CC BY 4.0](#)).¹¹

5 Summary

6 languagemodels is a Python package for educators and learners exploring the applications
7 of large language models. It aims to be as easy to set up and use as possible, while
8 providing many of the key building blocks used in modern LLM-driven applications. It is
9 designed to be used in learning modules in introductory programming courses.

Statement of Need

12 Large language models are having an impact on the way software is designed ([Mialon et al., 2023](#)). The development of the transformer ([Vaswani et al., 2017](#)) has led to rapid
13 progress in many NLP and generative tasks ([Brown et al., 2020](#); [Bubeck et al., 2023](#);
14 [Chowdhery et al., 2022](#); [Chung et al., 2022](#); [Devlin et al., 2018](#); [Radford et al., 2019](#); [Raffel et al., 2020](#); [Zhao et al., 2023](#)). These models are becoming more powerful as they scale in
15 both parameters ([Kaplan et al., 2020](#)) and training data ([Hoffmann et al., 2022](#)).
16

17 Early research suggests that there are many tasks performed by humans that can be
18 transformed by LLMs ([Eloundou et al., 2023](#)). For example, large language models trained
19 on code ([Chen et al., 2021](#)) are already being used as capable pair programmers via tools
20 such as Microsoft's Copilot. To build with these technologies, students need to understand
21 their capabilities and begin to learn new paradigms for programming.

22 There are many software tools already available for working with large language models
23 ([Abadi et al., 2015](#); [Anand et al., 2023](#); [Chase, 2022](#); [Gerganov, 2023](#); [Paszke et al., 2019](#);
24 [Wolf et al., 2020](#)). While these options serve the needs of software engineers, researchers,
25 and hobbyists, they may not be simple enough for new learners. This package aims to
26 lower the barriers to entry for using these tools in an educational context.

Example Usage

This package eliminates boilerplate and configuration options that create noise for new learners while using only basic types and simple functions. Here's an example from a Python REPL session:

```
>>> import languagemodels as lm

>>> lm.do("Answer the question: What is the capital of France?")
'Paris.'

>>> lm.do("Classify as positive or negative: I like games",
...       choices=["positive", "negative"])
'positive'

>>> lm.extract_answer("What color is the ball?",
...                   "There is a green ball and a red box")
'green'

>>> lm.get_wiki('Chemistry')
'Chemistry is the scientific study...'

>>> lm.store_doc(lm.get_wiki("Python"), "Python")
>>> lm.store_doc(lm.get_wiki("Javascript"), "Javascript")
>>> lm.get_doc_context("What language is used on the web?")
'From Javascript document: Javascript engines were...'
```

Features

Despite its simplicity, this package provides a number of building blocks that can be combined to build applications that mimic the architectures of modern software products. Some of the tools included are:

- Instruction following with the `do` function
- Zero-shot classification with the `do` function and `choices` parameter
- Semantic search using the `store_doc` and `get_doc_context` functions
- Extractive question answering using the `extract_answer` function
- Basic web retrieval using the `get_wiki` function

The package includes the following features under the hood:

- Local LLM inference on CPU for broad device support
- Transparent model caching to allow fast repeated inference without explicit model initialization
- Pre-selected models to allow the software to run easily and effectively on as many devices as possible

Implementation

The design of this software package allows its interface to be loosely coupled to the models and inference engines it uses. Progress is being made to speed up inference on consumer hardware, and this package seeks to find a balance between inference efficiency, software stability, and broad hardware support.

This package currently uses CTranslate2 (Klein et al., 2020) for efficient inference on CPU and GPU. The main models used include Flan-T5 (Chung et al., 2022), LaMini-LM (Wu et al., 2023), and OpenChat (Wang et al., 2023). The default models used by this package can be swapped out in future versions to provide improved generation quality.

Future work

This package provides a platform for creating simple NLP labs for use in introductory computer science courses. Additional work is needed to design specific learning modules to meet the needs of learners.

Ongoing development efforts will focus on improving the accuracy and efficiency of inference, while keeping the interface stable and supporting all reasonable platforms.

References

- Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., ... Zheng, X. (2015). *TensorFlow: Large-scale machine learning on heterogeneous systems*. <https://www.tensorflow.org/>
- Anand, Y., Nussbaum, Z., Duderstadt, B., Schmidt, B., & Mulyar, A. (2023). GPT4All: Training an assistant-style chatbot with large scale data distillation from GPT-3.5-turbo. In *GitHub repository*. <https://github.com/nomic-ai/gpt4all>; GitHub.
- Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., & others. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33, 1877–1901.
- Bubeck, S., Chandrasekaran, V., Eldan, R., Gehrke, J., Horvitz, E., Kamar, E., Lee, P., Lee, Y. T., Li, Y., Lundberg, S., & others. (2023). Sparks of artificial general intelligence; Early experiments with gpt-4. *arXiv Preprint arXiv:2303.12712*.
- Chase, H. (2022). LangChain: Building applications with LLMs through composability. In *GitHub*. <https://github.com/hwchase17/langchain>
- Chen, M., Tworek, J., Jun, H., Yuan, Q., Pinto, H. P. de O., Kaplan, J., Edwards, H., Burda, Y., Joseph, N., Brockman, G., & others. (2021). Evaluating large language models trained on code. *arXiv Preprint arXiv:2107.03374*.
- Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., & others. (2022). Palm: Scaling language modeling with pathways. *arXiv Preprint arXiv:2204.02311*.
- Chung, H. W., Hou, L., Longpre, S., Zoph, B., Tay, Y., Fedus, W., Li, E., Wang, X., Dehghani, M., Brahma, S., & others. (2022). Scaling instruction-finetuned language models. *arXiv Preprint arXiv:2210.11416*.
- Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv Preprint arXiv:1810.04805*.

- 88 Eloundou, T., Manning, S., Mishkin, P., & Rock, D. (2023). Gpts are gpts: An early
89 look at the labor market impact potential of large language models. *arXiv Preprint*
90 *arXiv:2303.10130*.
- 91 Gerganov, G. (2023). Llama.cpp: Port of facebook's llama model in c/c++. In *GitHub*.
92 <https://github.com/ggerganov/llama.cpp>
- 93 Hoffmann, J., Borgeaud, S., Mensch, A., Buchatskaya, E., Cai, T., Rutherford, E., Casas,
94 D. de L., Hendricks, L. A., Welbl, J., Clark, A., & others. (2022). Training compute-
95 optimal large language models. *arXiv Preprint arXiv:2203.15556*.
- 96 Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S.,
97 Radford, A., Wu, J., & Amodei, D. (2020). Scaling laws for neural language models.
98 *arXiv Preprint arXiv:2001.08361*.
- 99 Klein, G., Hernandez, F., Nguyen, V., & Senellart, J. (2020). The OpenNMT neural
100 machine translation toolkit: 2020 edition. *Proceedings of the 14th Conference of the*
101 *Association for Machine Translation in the Americas (Volume 1: Research Track)*,
102 102–109.
- 103 Mialon, G., Dessì, R., Lomeli, M., Nalmpantis, C., Pasunuru, R., Raileanu, R., Rozière, B.,
104 Schick, T., Dwivedi-Yu, J., Celikyilmaz, A., & others. (2023). Augmented language
105 models: A survey. *arXiv Preprint arXiv:2302.07842*.
- 106 Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen,
107 T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E.,
108 DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L.,
109 ... Chintala, S. (2019). PyTorch: An imperative style, high-performance deep
110 learning library. In *Advances in neural information processing systems 32*
111 (pp. 8024–8035). Curran Associates, Inc. [http://papers.neurips.cc/paper/](http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf)
112 [9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf](http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf)
- 113 Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., & others. (2019).
114 Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8), 9.
- 115 Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., &
116 Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text
117 transformer. *The Journal of Machine Learning Research*, 21(1), 5485–5551.
- 118 Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł.,
119 & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information*
120 *Processing Systems*, 30.
- 121 Wang, G., Cheng, S., Zhan, X., Li, X., Song, S., & Liu, Y. (2023). OpenChat:
122 Advancing open-source language models with mixed-quality data. *arXiv Preprint*
123 *arXiv:2309.11235*.
- 124 Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T.,
125 Louf, R., Funtowicz, M., Davison, J., Shleifer, S., Platen, P. von, Ma, C., Jernite, Y., Plu,
126 J., Xu, C., Scao, T. L., Gugger, S., ... Rush, A. M. (2020). *HuggingFace's transformers:*
127 *State-of-the-art natural language processing*. <https://arxiv.org/abs/1910.03771>
- 128 Wu, M., Waheed, A., Zhang, C., Abdul-Mageed, M., & Aji, A. F. (2023). LaMini-LM: A
129 diverse herd of distilled models from large-scale instructions. *CoRR*, *abs/2304.14402*.
130 <https://arxiv.org/abs/2304.14402>
- 131 Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang,
132 J., Dong, Z., & others. (2023). A survey of large language models. *arXiv Preprint*
133 *arXiv:2303.18223*.