What is LIFT

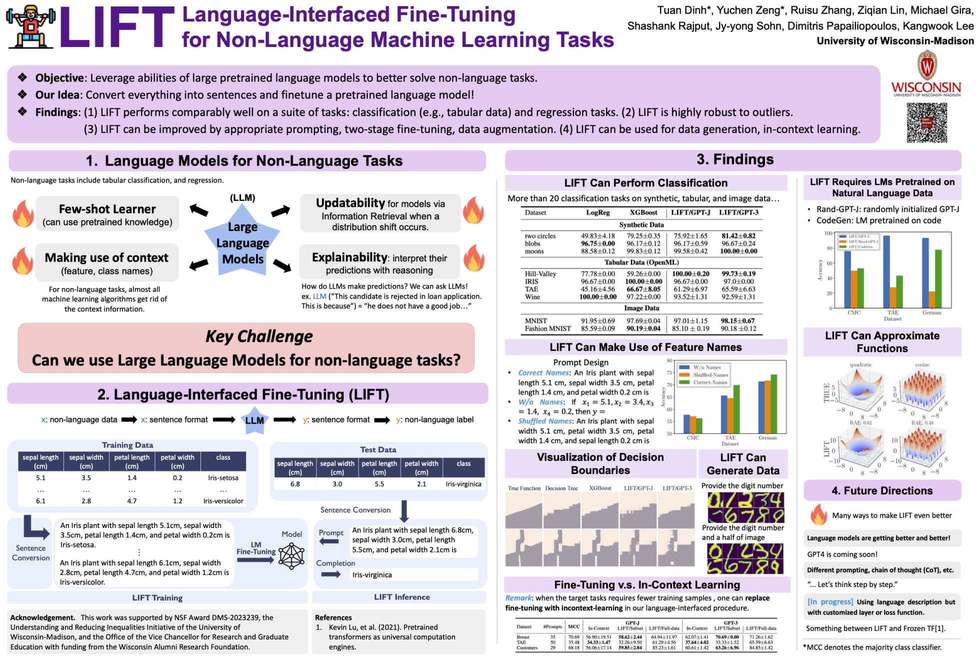
Language-Interfaced Fine-Tuning for Non-Language Machine Learning Tasks

1. Background:

Pre-trained transformer language models have shown great success in natural language processing and protein/molecule design. This study explores their application in generating inorganic material compositions, a novel approach with potential for high-throughput materials discovery.

2. Challenge:

Navigating the vast design space of inorganic materials for new discoveries is costly and challenging. Pre-trained transformer models are leveraged to address the complexity of material composition generation, aiming for chemically valid and structurally stable results.



LIFT Language-Interfaced Fine-Tuning for Non-Language Machine Learning Tasks

Tuan Dinh*, Yuchen Zeng*, Ruisu Zhang, Ziqian Lin, Michael Gira,
Shashank Rajput, Jy-yong Sohn, Dimitris Papailiopoulos, Kangwook Lee
University of Wisconsin-Madison

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

- **Objective**: Leverage abilities of large pretrained language models to better solve non-language tasks.
- **Our Idea**: Convert everything into sentences and finetune a pretrained language model!
- **Findings**: (1) LIFT performs comparably well on a suite of tasks: classification (e.g., tabular data) and regression tasks. (2) LIFT is highly robust to outliers. (3) LIFT can be improved by appropriate prompting, two-stage fine-tuning, data augmentation. (4) LIFT can be used for data generation, in-context learning.

1. Language Models for Non-Language Tasks

Non-language tasks include tabular classification, and regression.

(LLM)

Large Language Models

Few-shot Learner (can use pretrained knowledge)

Making use of context (feature, class names)

For non-language tasks, almost all machine learning algorithms get rid of the context information.

Updatability for models via Information Retrieval when a distribution shift occurs.

Explainability: interpret their predictions with reasoning

How do LLMs make predictions? We can ask LLMs! ex. LLM (“This candidate is rejected in loan application. This is because”) = “he does not have a good job…”

Key Challenge

Can we use Large Language Models for non-language tasks?

2. Language-Interfaced Fine-Tuning (LIFT)

x: non-language data → x: sentence format → LLM → y: sentence format → y: non-language label

Training Data

sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	class
5.1	3.5	1.4	0.2	Iris-setosa
…	…	…	…	…
6.1	2.8	4.7	1.2	Iris-versicolor

Test Data

sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	class
6.8	3.0	5.5	2.1	Iris-virginica

Sentence Conversion

An Iris plant with sepal length 5.1cm, sepal width 3.5cm, petal length 1.4cm, and petal width 0.2cm is Iris-setosa.

⋮

An Iris plant with sepal length 6.1cm, sepal width 2.8cm, petal length 4.7cm, and petal width 1.2cm is Iris-versicolor.

LM Fine-Tuning → Model

LIFT Training

Sentence Conversion

Prompt: An Iris plant with sepal length 6.8cm, sepal width 3.0cm, petal length 5.5cm, and petal width 2.1cm is

Completion: Iris-virginica

LIFT Inference

Acknowledgement. This work was supported by NSF Award DMS-2023239, the Understanding and Reducing Inequalities Initiative of the University of Wisconsin-Madison, and the Office of the Vice Chancellor for Research and Graduate Education with funding from the Wisconsin Alumni Research Foundation.

References

1. Kevin Lu, et al. (2021). Pretrained transformers as universal computation engines.

3. Findings

LIFT Can Perform Classification

More than 20 classification tasks on synthetic, tabular, and image data…

Dataset	LogReg	XGBoost	LIFT/GPT-J	LIFT/GPT-3
Synthetic Data				
two circles	49.83±4.18	79.25±0.35	75.92±1.65	**81.42±0.82**
blobs	**96.75±0.00**	96.17±0.12	96.17±0.59	96.67±0.24
moons	88.58±0.12	99.83±0.12	99.58±0.42	**100.00±0.00**
Tabular Data (OpenML)				
Hill-Valley	77.78±0.00	59.26±0.00	**100.00±0.20**	**99.73±0.19**
IRIS	96.67±0.00	**100.00±0.00**	96.67±0.00	97.0±0.00
TAE	45.16±4.56	**66.67±8.05**	61.29±6.97	65.59±6.63
Wine	**100.00±0.00**	97.22±0.00	93.52±1.31	92.59±1.31
Image Data				
MNIST	91.95±0.69	97.69±0.04	97.01±1.15	**98.15±0.67**
Fashion MNIST	85.59±0.09	**90.19±0.04**	85.10 ± 0.19	90.18 ±0.12

LIFT Can Make Use of Feature Names

Prompt Design

- *Correct Names*: An Iris plant with sepal length 5.1 cm, sepal width 3.5 cm, petal length 1.4 cm, and petal width 0.2 cm is
- *W/o Names*: If $x_1 = 5.1, x_2 = 3.4, x_3 = 1.4, x_4 = 0.2$, then $y =$
- *Shuffled Names*: An Iris plant with sepal width 5.1 cm, petal width 3.5 cm, petal width 1.4 cm, and sepal length 0.2 cm is

Visualization of Decision Boundaries

LIFT Can Generate Data

Provide the digit number

Provide the digit number and a half of image

Fine-Tuning v.s. In-Context Learning

Remark: when the target tasks requires fewer training samples , one can **replace fine-tuning with incontext-learning** in our language-interfaced procedure.

Dataset	#Prompts	MCC	GPT-J In-Context	GPT-J LIFT/Subset	GPT-J LIFT/Full-data	GPT-3 In-Context	GPT-3 LIFT/Subset	GPT-3 LIFT/Full-data
Breast	35	70.69	56.90±19.51	**58.62±2.44**	64.94±11.97	62.07±1.41	**70.69±0.00**	71.26±1.62
TAE	50	35.48	**34.33±1.47**	32.26±9.50	61.29±4.56	**37.64±4.02**	33.33±1.52	65.59±6.63
Customers	29	68.18	56.06±17.14	**59.85±2.84**	85.23±1.61	60.61±1.42	**63.26±6.96**	84.85±1.42

LIFT Requires LMs Pretrained on Natural Language Data

- Rand-GPT-J: randomly initialized GPT-J
- CodeGen: LM pretrained on code

LIFT Can Approximate Functions

4. Future Directions

Many ways to make LIFT even better

Language models are getting better and better!

GPT4 is coming soon!

Different prompting, chain of thought (CoT), etc.

“... Let’s think step by step.”

[In progress] **Using language description but with customized layer or loss function.**

Something between LIFT and Frozen TF[1].

*MCC denotes the majority class classifier.

3. Current Status:
   The study demonstrates the effectiveness of transformer models in generating hypothetical material compositions. Training different transformer models on diverse datasets yields high validity percentages, uniqueness, and potential for new materials discovery, opening new possibilities for materials design.

4. Method:

   Seven transformer language models, including GPT, GPT-2, and RoBERTa, are trained on various material datasets. Validity, uniqueness, recovery rate, and novelty are evaluated to assess generative performance. Hyperparameters are tuned, and formation energies are predicted using DFT calculations.

5. Result:

   Transformer-based models exhibit high validity and novelty in generating material compositions. MT-GPTJ shows the best performance, closely followed by MT-GPT2 and MT-GPTNeo. MT-BART and MT-RoBERTa lag slightly behind. Training set size and dataset quality significantly impact model performance.

6. Conclusion:

   Transformer language models show promise in generative materials design, capable of learning chemical patterns from training datasets. Despite biases and challenges in generating specific compositions, these models offer a new approach for materials exploration and potential discovery.

7. Outlook:
   Future research may focus on addressing biases in composition generation, optimizing model training on diverse datasets, and integrating crystal structure predictions for synthesized materials. Continued development of transformer models for materials design could lead to significant advancements in high-throughput materials discovery.

Thank you!