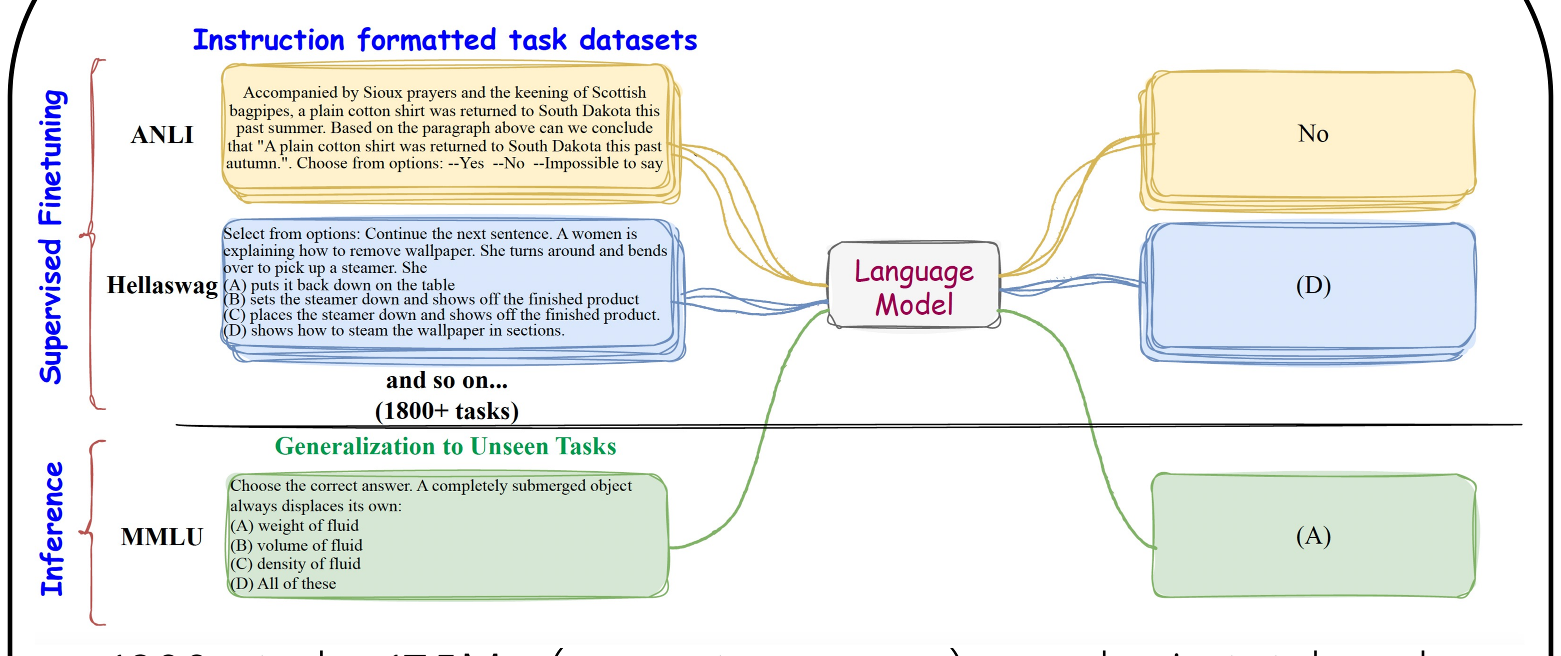
SMART: Submodular Data Mixture Strategy for Instruction Tuning

H S V N S Kowndinya Renduchintala[1], Sumit Bhatia[1], Ganesh Ramakrishnan[2]

Research Questions



1800+ tasks; 17.5M+ (prompt, response) samples in total; and a limited finetuning budget of only - say 100,000 samples?? 😕

How many samples to select from each task? And which samples?🤔

And, do we really need all tasks? 😅

May be only a few representative tasks are enough... 🤷

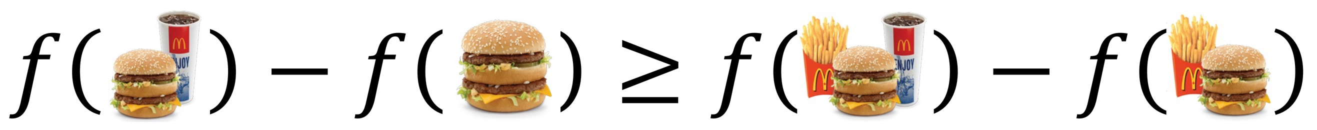
Submodular Functions

A set function $f: 2^{\mathcal{V}} \rightarrow \mathbb{R}$ is called a submodular function if the following diminishing gains property is satisfied:

$$f(X \cup \{v\}) - f(X) \geq f(Y \cup \{v\}) - f(Y)$$

$$\forall\, X \subseteq Y \subseteq \mathcal{V};\ v \in \mathcal{V} \setminus Y$$

Example – Consumer costs are typically submodular:



Examples of Submodular Functions

Submodular Function	$f(X)$
Facility Location	$\sum_{i \in \mathcal{V}} \max_{j \in X} s_{ij}$
Graph Cut	$\sum_{i \in \mathcal{V}, j \in X} s_{ij} - \lambda \sum_{i,j \in X} s_{ij}$
Log Determinant	$\log\det(\mathcal{S}_X)$

($\mathcal{V}$ is the ground set and $X \subseteq \mathcal{V}$)

s_{ij} is the similarity between two elements i and j of the ground set and $\mathcal{S}_X$ is the similarity matrix between items in X

Facility Location models representation; Log Determinant models diversity; Graph Cut models a trade-off between representation and diversity controlled by the parameter λ.

Cardinality Constrained Submodular Maximization

$$S^* = arg \max_{\substack{X \subseteq \mathcal{V} \\ |X| \leq N'}} f(X)$$

Algorithm 1 The Naïve Greedy

```
Input: Ground Set (V), Budget (N')
  X_0 ← ∅;
  S ← [];
  Gains ← [];
for i = 0 to (N' − 1) do
    e* = argmax_{v∈V\X_i} f(v|X_i);
    g_{i+1} = f(e*|X_i);
    X_{i+1} = X_i ∪ {e*};
    S.append(e*);
    Gains.append(g_{i+1});
end for
return S, Gains;
```

This is NP-complete in general. But if f is monotone submodular, then a simple greedy algorithm can be used to find an approximate solution with the following guarantee:

$$f(S^{greedy}) \geq \left(1 - \frac{1}{e}\right) f(S^*)$$

The SMART Algorithm

Let's say we have a collection of M instruction-formatted task datasets - $\mathcal{D} = \{T_1, T_2, \ldots, T_M\}$, where each $T_i = \{(prompt_{ij}, response_{ij})\}_{j=1}^{N_{T_i}}$ consists of N_{T_i} (prompt, response) pairs such that $\sum_{i=1}^{M} N_{T_i} = N$.

Given an $M' \leq M$ and an $N' \leq N$, how do we select a subset of M' tasks $\mathcal{D}' = \{T_1', T_2', \ldots, T_{M'}'\}$ ($\mathcal{D}' \subseteq \mathcal{D}$), and subsequently $\mathcal{S} = \{S_1, S_2, \ldots, S_{M'}\}$, where $S_j \subseteq T_j$ and $\sum_{j=1}^{M'} |S_j| = N'$, such that fine-tuning on $\mathcal{S}$ alone is (nearly) as effective as fine-tuning on the entire $\mathcal{D}$?

Stage-1: Weighted Task Subset Selection

$$\mathcal{D}' = \arg\max_{\substack{X \subseteq \mathcal{D} \\ |X| \leq M'}} f_1(X)$$

If $\{g_1, g_2, \ldots, g_{M'}\}$ are the corresponding value gains, then the task budgets are computed as

$$N_j' = \frac{(1 + g_j + 0.5 g_j^2)}{\sum_{k=1}^{M'} (1 + g_k + 0.5 g_k^2)} N_j$$

Stage-2: Instance Subset Selection

$$\mathcal{S} = \bigcup_{j=1}^{M'} \arg\max_{\substack{X_j \subseteq T_j' \\ |X_j| \leq N_j'}} f_2(X)$$

Choosing (f_1, f_2)

A Grid Search revealed that:

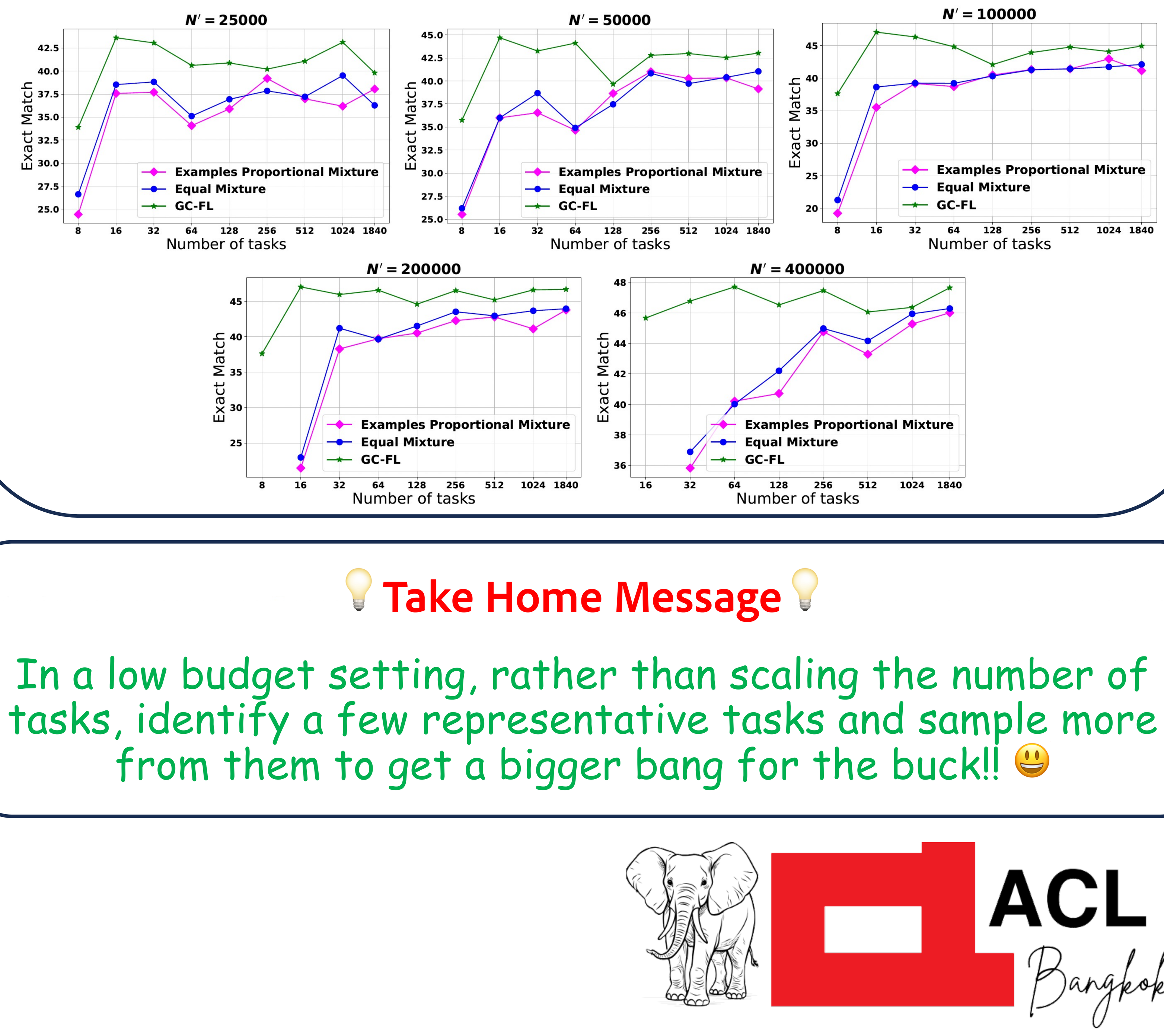
- Graph Cut works best for f_1
- The optimal f_2 however, also depends on number of tasks (M') – For higher M's, each task on average gets a relatively low budget and the need for representation dominates the need for diversity; however, when there is sufficient budget for each tasks (lower M's), the need for diversity takes over.

$M' = M$

		MMLU-ZeroShot (Exact Match)					BBH-Zeroshot (Exact Match)			MMLU + BBH
N'	Data Mix.	STEM	Humanities	Social Sciences	Other	MMLU FULL	NLP	Algorithmic	BBH FULL	(Weighted Avg.)
25000	EPM (Baseline-1)	30.82	46	46.71	43.05	40.63	**40.59**	**25.55**	**31.67**	38.05
	EM (Baseline-2)	30.33	45.01	43.81	40.55	39.03	40.08	20.29	29.46	36.27
	SMART (Ours)	**32.22**	**50.41**	**50.14**	**46.85**	**43.73**	38.85	24.16	30.05	**39.8**
50000	EPM (Baseline-1)	31.59	47.68	47.18	44.68	41.76	41.25	**26.49**	**32.64**	39.14
	EM (Baseline-2)	35.22	49.58	51.01	48.13	44.99	41.96	22.83	31.24	41.04
	SMART (Ours)	**36.51**	**53.06**	**54.49**	**50.79**	**47.58**	**46.73**	20.1	31.75	**43.03**
100000	EPM (Baseline-1)	32.66	50.6	51.37	47.18	44.25	43.38	**26.36**	33.48	41.16
	EM (Baseline-2)	36.03	50.7	52.53	47.1	45.57	44.4	25.18	33.55	42.11
	SMART (Ours)	**37.36**	**55.38**	**55.47**	**52.95**	**49.11**	**47.26**	24.22	**34.66**	**44.96**
200000	EPM (Baseline-1)	35.19	54.64	54.75	50.58	47.53	45.25	**26.57**	34.52	43.79
	EM (Baseline-2)	38.6	54.05	54.72	51.47	48.68	41.36	24.75	32.28	43.96
	SMART (Ours)	**39.2**	**57.29**	**58.71**	**55.01**	**51.32**	**47.99**	24.47	**35.27**	**46.7**
400000	EPM (Baseline-1)	38.16	56.53	56.99	52.56	49.85	48.72	26.04	36.49	46.01
	EM (Baseline-2)	39.43	55.97	57.59	53.65	50.52	47.37	26.08	35.8	46.29
	SMART (Ours)	**39.77**	**57.39**	**60.17**	**54.79**	**51.77**	**49.25**	**26.35**	**37.43**	**47.65**
17,591,640	Full FLAN 2022	42.44	59.1	61.82	55.1	53.43	50.7	27.6	38.11	49.03

$M' < M$

(Weighted avg. of exact matches on MMLU-zeroshot and BBH-zeroshot)



💡 Take Home Message 💡

In a low budget setting, rather than scaling the number of tasks, identify a few representative tasks and sample more from them to get a bigger bang for the buck!! 😃

[1]Adobe Inc., India
[2]Indian Institute of Technology Bombay, India

ACL 2024
Bangkok, Thailand