
北京大学
PEKING UNIVERSITY

# Instance Selection for In-context Learning

Kun-Peng Ning

# Motivation

**What Makes Good In-Context Examples for GPT-3?**

**Jiachang Liu**[1]*, **Dinghan Shen**[2], **Yizhe Zhang**[3], **Bill Dolan**[3], **Lawrence Carin**[1], **Weizhu Chen**[2]
[1]Duke University [2]Microsoft Dynamics 365 AI [3]Microsoft Research
[1]{jiachang.liu, lcarin}@duke.edu
[2,3]{dishen, yizzhang, billdol, wzchen}@microsoft.com

ACL-W, 2021, citation 621

# Motivation

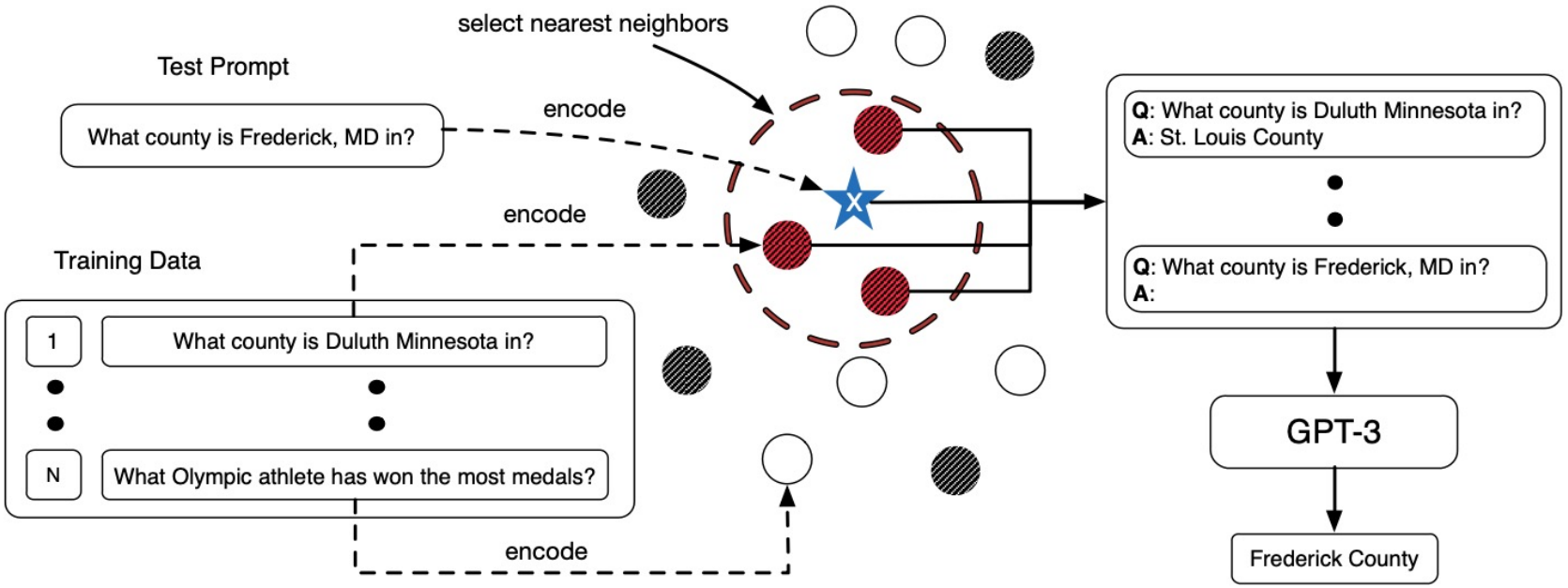



Figure 2: In-context example selection for GPT-3. White dots: unused training samples; grey dots: randomly sampled training samples; red dots: training samples selected by the $k$-nearest neighbors algorithm in the embedding space of a sentence encoder.

RAG （Retrieval-Augmented Generation）

KATE: Large labeled set -> Small labeled set -> Prompt

# Motivation

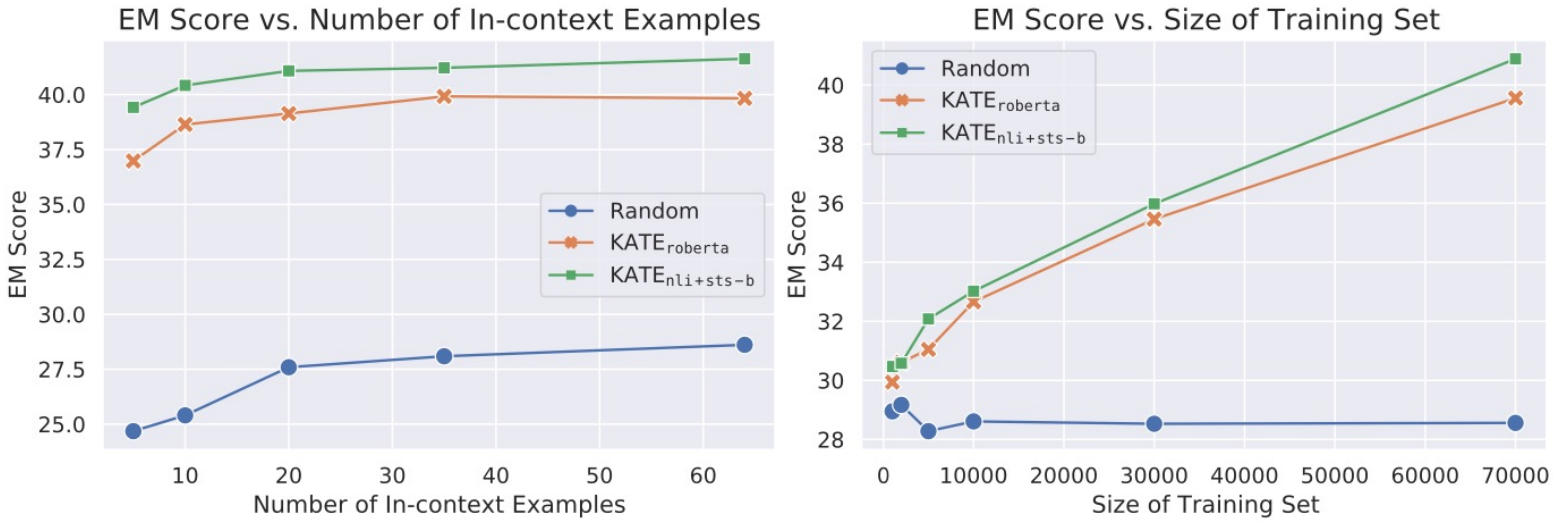



Figure 3: Left: Ablation study on the effect of number of in-context examples for GPT-3 for different selection methods. Right: Ablation study on the effect of the size of training set for retrieval on KATE. Two representative sentence encoders are used in the ablation study.

- Carefully designed instance selection is effective.
- The larger the candidate(knowledge) set, the better the performance.
- Supervised scenario.

# SELECTIVE ANNOTATION MAKES LANGUAGE MODELS BETTER FEW-SHOT LEARNERS

**Hongjin Su**[♠] **Jungo Kasai**[♣◇] **Chen Henry Wu**[♡] **Weijia Shi**[♣] **Tianlu Wang**[♦] **Jiayi Xin**[♠]
**Rui Zhang**[★] **Mari Ostendorf**[♣] **Luke Zettlemoyer**[♣♦] **Noah A. Smith**[♣◇] **Tao Yu**[♠♣]
[♠]The University of Hong Kong [♣]University of Washington [◇]Allen Institute for AI
[♡]Carnegie Mellon University [★]Penn State University [♦]Meta AI

{hjsu,tyu}@cs.hku.hk, henrychenwu@cmu.edu, ostendor@uw.edu
{jkasai,swj0419,lsz,nasmith}@cs.washington.edu

ICLR'23, citation 86

# Vote-k

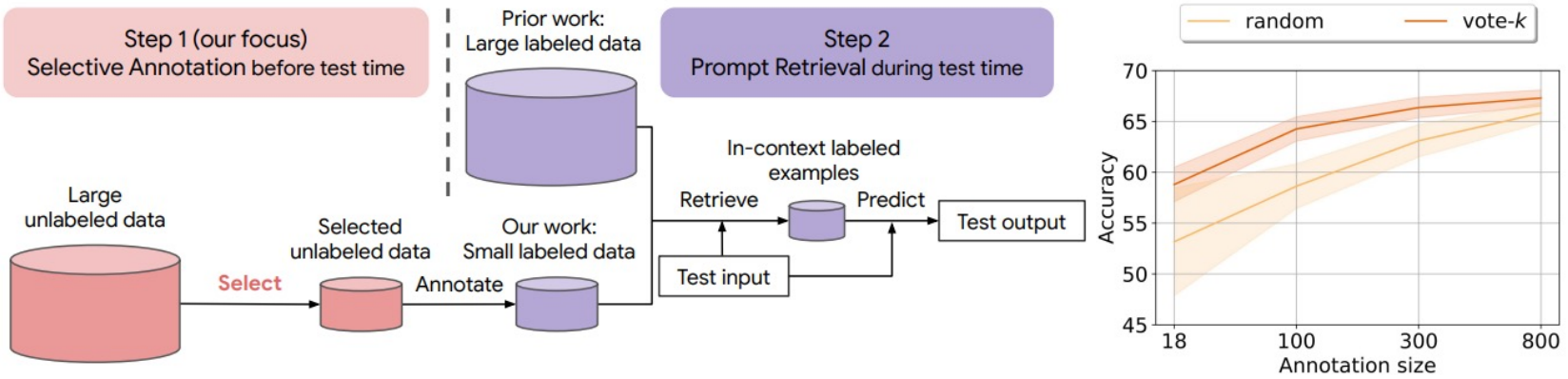
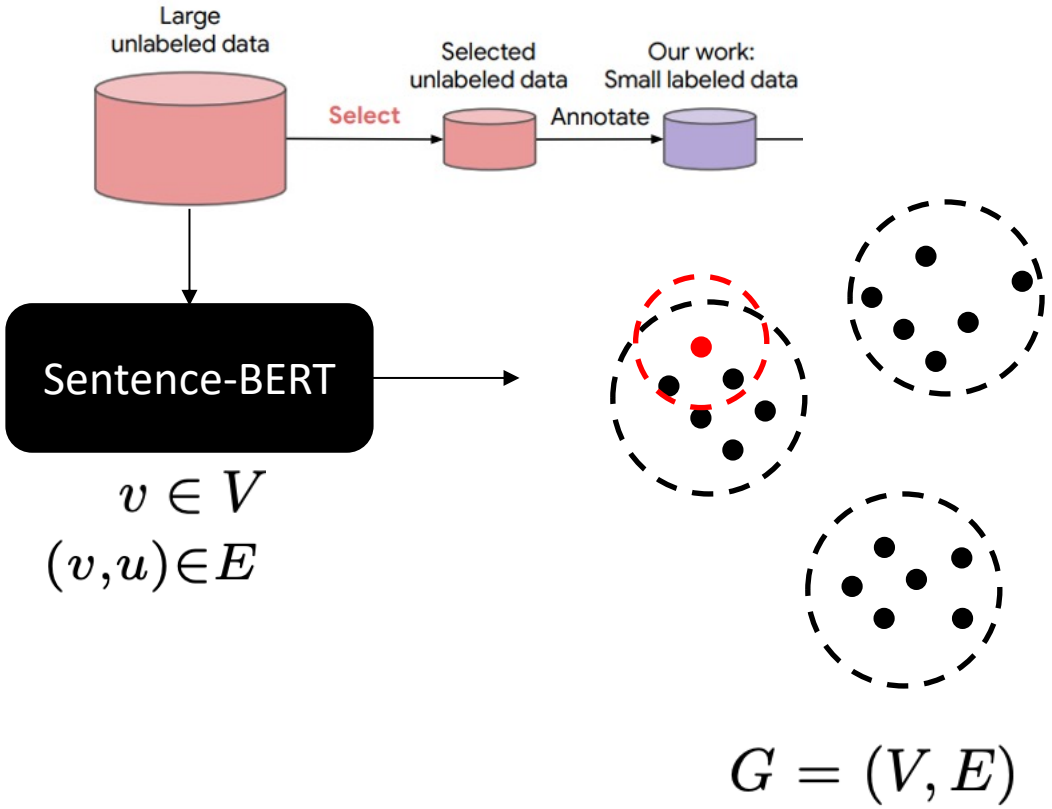



Figure 1: **Left**: Our two-step framework for in-context learning. Instead of assuming access to large labeled data, we first select a small number of (diverse and representative) unlabeled examples to annotate before test time. At test time, we retrieve in-context examples from the small annotated pool. **Right**: In-context learning performance over varying annotation budgets averaged over three representative tasks (HellaSwag commonsense reasoning, MRPC paraphrase detection, and MWOZ dialogue state tracking). Here we experiment with GPT-J and Codex-davinci-002. Two selective annotation methods are presented: *random selection* and our *vote-k* method. We observe that an appropriate selective annotation method largely improves the in-context learning performance with smaller variance over random selection under varying annotation budgets.

- Unsupervised scenario.
- Two-step framework.
- Vote-k selective annotation yields similar performance to state-of-the-art supervised finetuning with 10-100×less annotation cost.

# Vote-k

- Combine diversity and representativeness




$$\text{score}(u) = \sum_{v \in \{v | (v,u) \in E, v \in \mathcal{U}\}} s(v), \quad \text{where } s(v) = \rho^{-|\{\ell \in \mathcal{L} | (v,\ell) \in E\}|}, \quad \rho > 1$$

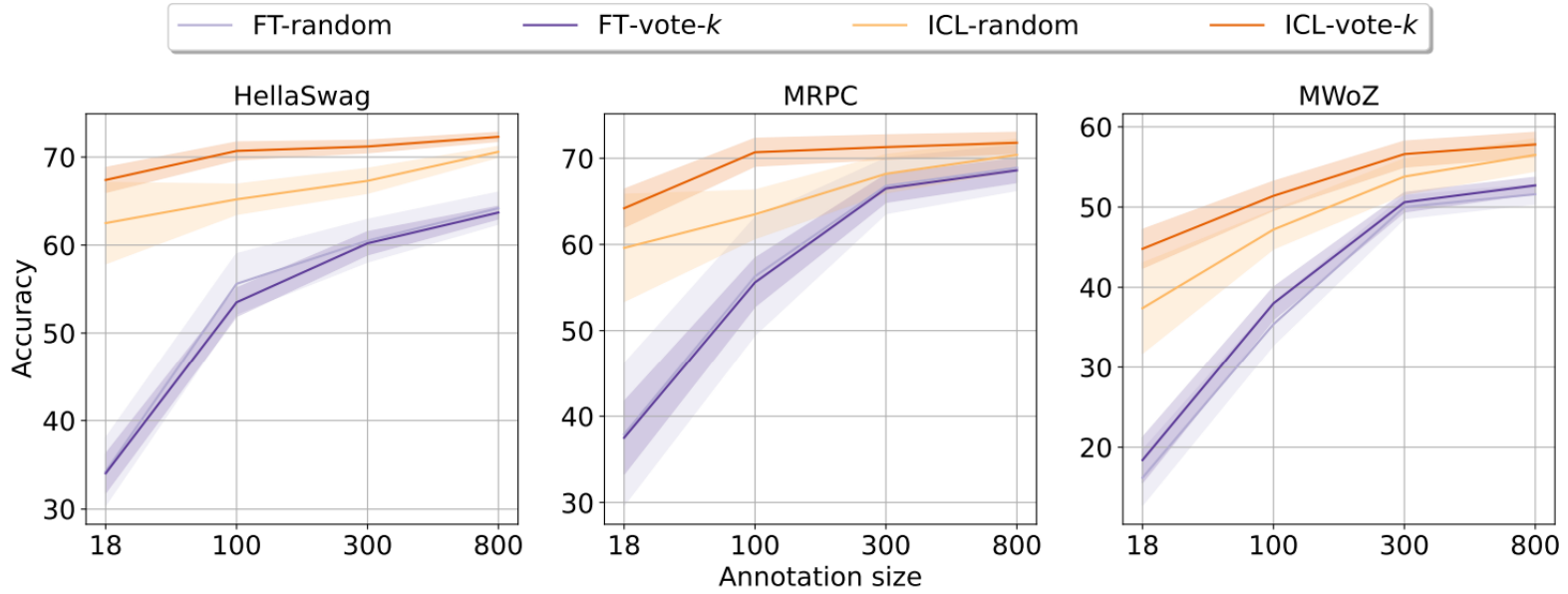
# Vote-k

**Algorithm 1** Voke-k Selective Annotation

**Input:** $\mathcal{X} = \{x_i\}_{i=1}^{N}$: a set of unlabeled samples; $M$: the number of samples to be selected; LM: inference language model.
**Initialization:** $\mathcal{L} = \varnothing, \mathcal{U} = \mathcal{X}$. $G = (V, E)$, where $V = \mathcal{X}$ and $(u, v) \in E$ if $v$ is one of $u$'s $k$ nearest vertices in terms of the cosine similarity between the embeddings.
**while** $|\mathcal{L}| < M/10$ **do**
    $u^* = \arg\max_{u \in \mathcal{U}} \sum_{v \in \{v|(v,u) \in E, v \in \mathcal{U}\}} s(v)$, where $s(v) = \rho^{-|\{\ell \in \mathcal{L} | (v,\ell) \in E\}|}$, $\rho > 1$
    $\mathcal{L} = \mathcal{L} \cup \{u^*\}$
    $\mathcal{U} = \mathcal{U} \setminus \{u^*\}$
**end while**
**for** $u$ in $\mathcal{U}$ **do**
    $\text{score}(u) = \frac{1}{\mathbf{q}} \sum_t \log p(q_t | \mathbf{q}_{<t}, \mathbf{z}; \Theta)$, where $p$ is LM prediction function and $\Theta$ is LM parameters
**end for**
**for** $j = 1, \ldots, 9$ **do**
    $\mathcal{U}_j = \text{indices}[(j-1)|\mathcal{U}|/10 : j|\mathcal{U}|/10]$
    **for** $i = 1, \ldots, |\mathcal{U}_j|$ **do**
        $u^* = \arg\max_{u \in \mathcal{U}_j} \sum_{v \in \{v|(v,u) \in E, v \in \mathcal{U}_j\}} s(v)$, where $s(v) = \rho^{-|\{\ell \in \mathcal{L} | (v,\ell) \in E\}|}$, $\rho > 1$
        $\mathcal{L} = \mathcal{L} \cup \{u^*\}$
        $\mathcal{U}_j = \mathcal{U}_j \setminus \{u^*\}$
    **end for**
**end for**
**Return:** $\mathcal{L}$: selected samples.

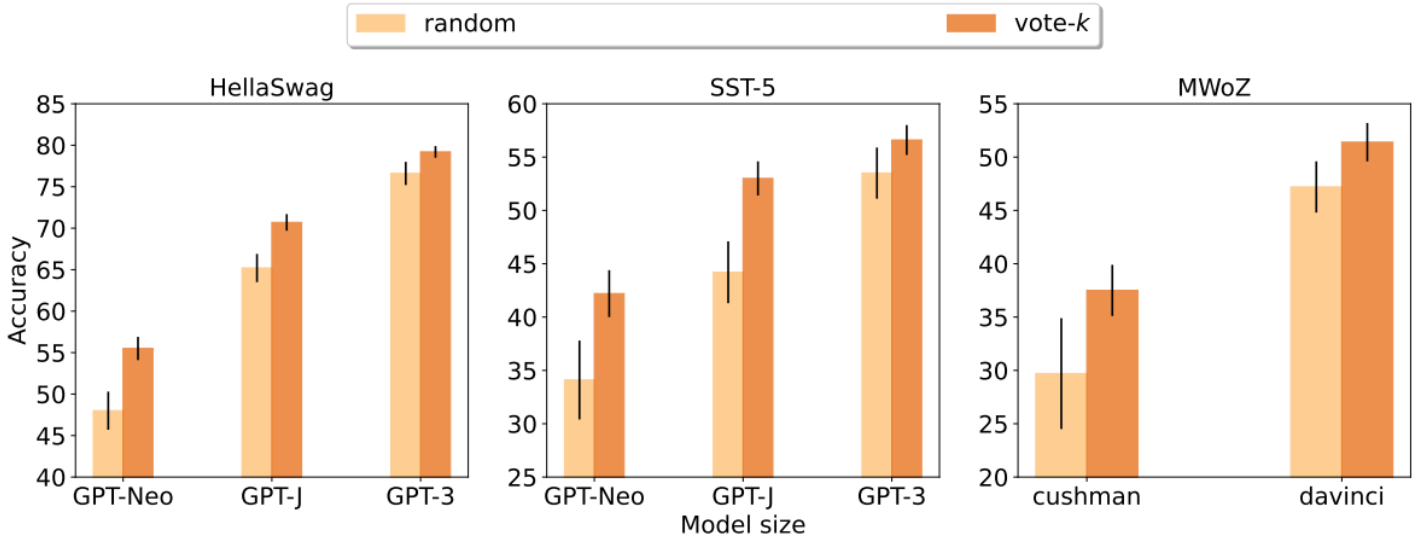
# Vote-k

| Method | | Classification | | | | | Multi-Choice | Dialogue | Generation | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $\|\mathcal{L}\|$ | Selection | MRPC | SST-5 | MNLI | DBpedia | RTE | HSwag | MWoZ | GeoQ | NQ | XSum |
| 100 | Random | 63.5 | 44.2 | 37.4 | 89.8 | 51.5 | 65.2 | 47.2 | 78.6 | 30.8 | 15.3 |
| 100 | Vote-*k* | **70.7** | **53.0** | **47.3** | **93.4** | **55.5** | **70.7** | **51.4** | **82.8** | **33.6** | **17.2** |
| 100 | Δ Absolute gain | +7.2 | +8.8 | +9.9 | +3.6 | +4.0 | +5.5 | +4.2 | +4.2 | +2.8 | +1.9 |
| 18 | Random | 59.6 | 39.8 | 36.7 | 77.6 | 50.4 | 62.5 | 33.6 | 62.4 | 29.8 | 13.6 |
| 18 | Vote-*k* | **64.2** | **47.6** | **41.0** | **87.1** | **54.3** | **67.4** | **42.8** | **72.5** | **32.3** | **15.2** |
| 18 | Δ Absolute gain | +4.8 | +7.8 | +4.3 | +9.5 | +3.9 | +4.9 | +8.8 | +9.9 | +2.5 | +1.6 |



# Vote-k

## 4.2 Language Models with Various Sizes




## 4.5 Alternative Selective Annotation Methods

| | Random | MFL | Diversity | Least-confidence | Fast vote-$k$ | Vote-$k$ |
|---|---|---|---|---|---|---|
| HellaSwag | 65.2 | 66.5 | 68.2 | 68.4 | 69.5 | **70.7** |
| SST-5 | 44.2 | 45.6 | 48.5 | 46.2 | 51.9 | **53.0** |
| MWoZ | 47.2 | 48.3 | 49.2 | 49.4 | 50.2 | **51.4** |

# Active Learning Principles for In-Context Learning with Large Language Models

**Katerina Margatina**[◇*] **Timo Schick**[†] **Nikolaos Aletras**[◇] **Jane Dwivedi-Yu**[†]
[◇]University of Sheffield [†]Meta AI Research
{k.margatina, n.aletras}@sheffield.ac.uk
{janeyu,schick}@meta.com

EMNLP'2023, citation 8

# Active Learning Principles for ICL

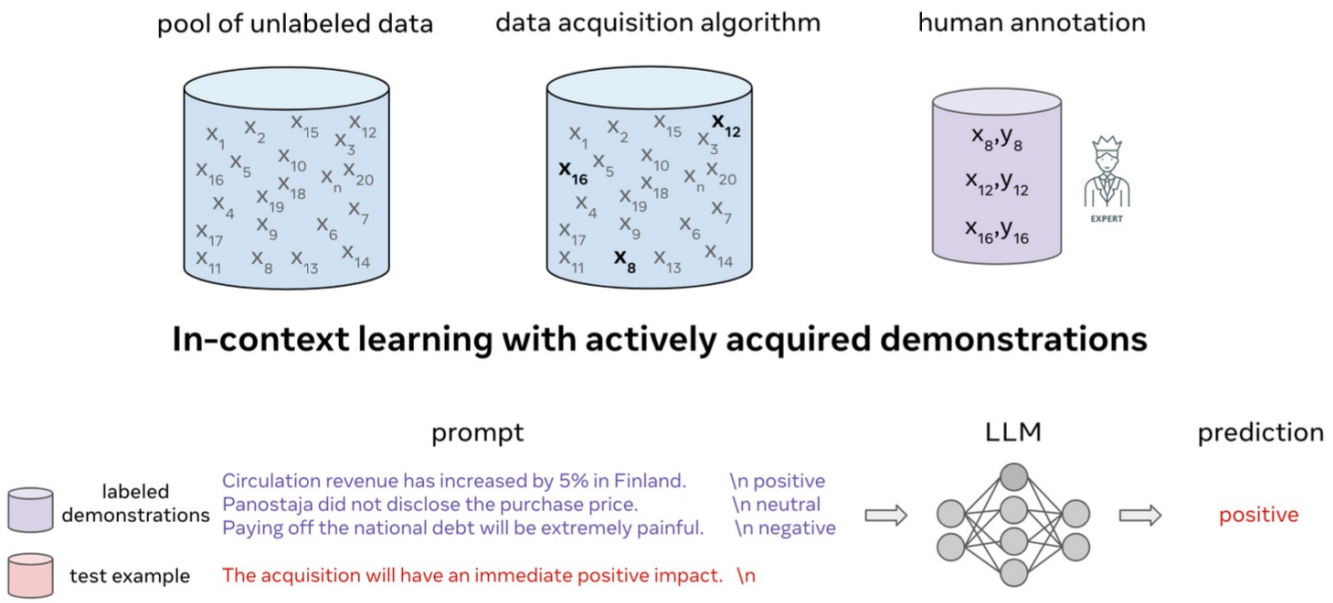



Figure 2: Top: Active data collection (single iteration). Bottom: Prompt construction and model inference.

# Active Learning Principles for ICL

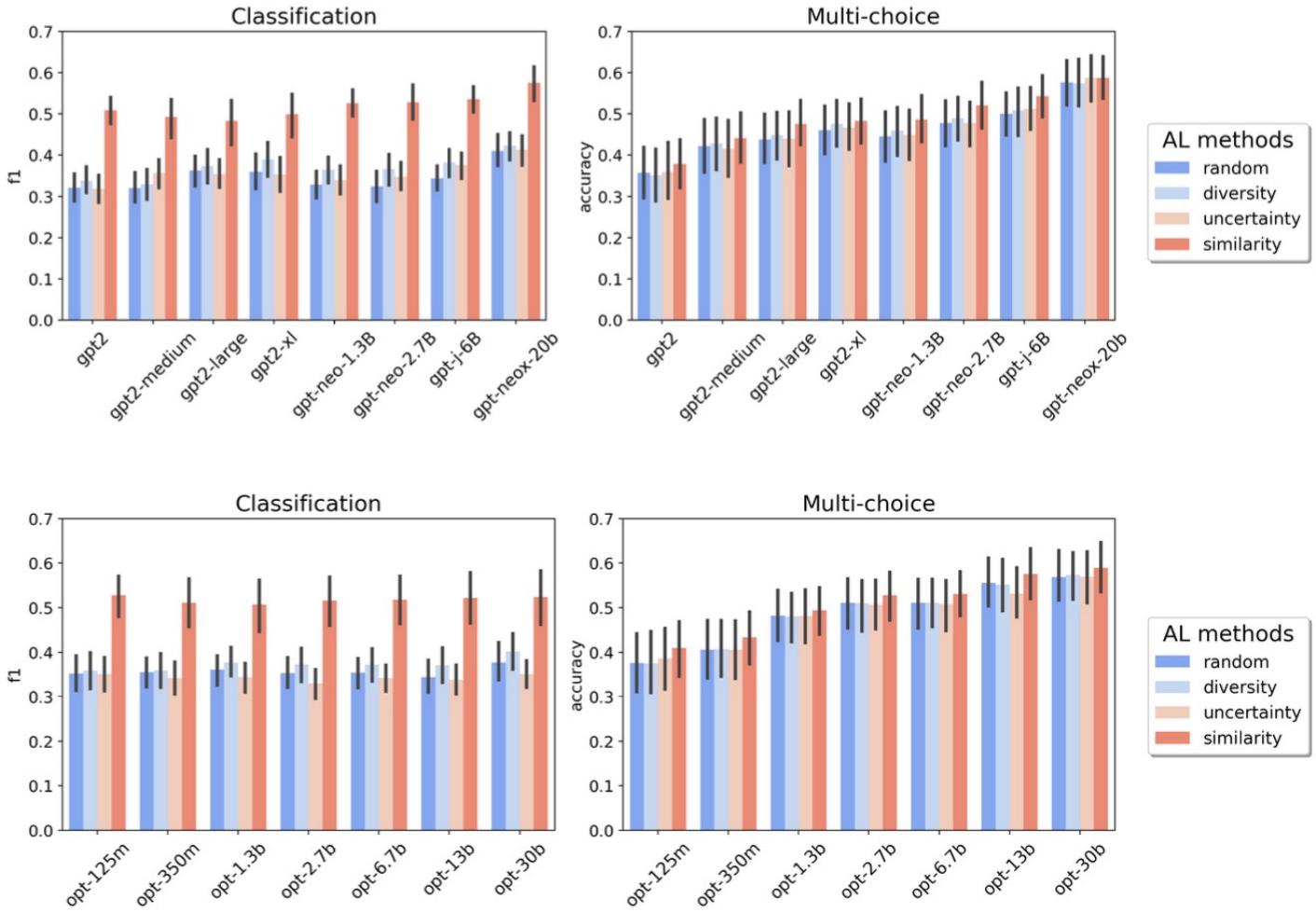



Figure 3: Results for various GPT (top) and OPT (bottom) models and AL methods averaged over 15 classification and 9 multi-choice tasks. *Similarity* is consistently the best performing approach overall, followed by *diversity* and *random*. Interestingly, we observe that *uncertainty* sampling underperforms in this setting of in-context learning.

# Active Learning Principles for ICL

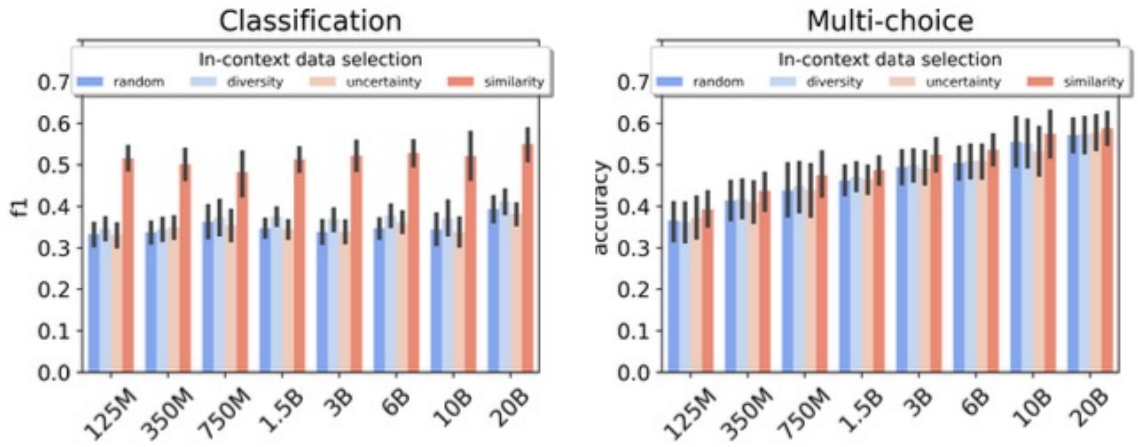



Figure 4: Results per model size.

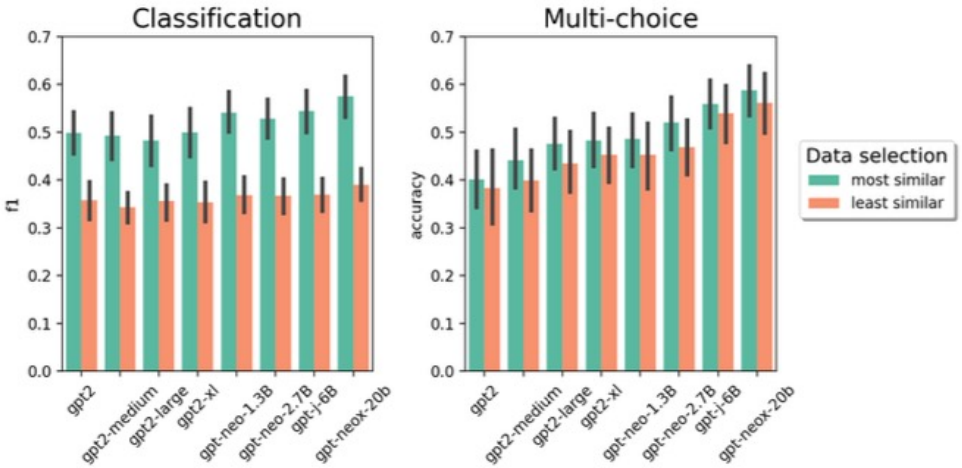



Figure 6: Most vs. least similar in-context examples.

**Similarity** Finally, the third active learning algorithm we consider is based on KATE a knn-augmented in-context example selection method proposed by Liu et al. (2022). The algorithm retrieves examples from the pool that are semantically-similar to a test query sample. We again use Sentence-BERT (Reimers and Gurevych, 2019) representations of both the pool and the test set for k nearest neighbours. The rationale behind this approach is that the most similar demonstrations to the test example will best help the model answer the query. We have to highlight, however, that by definition each test example will have a different prompt, as the $k$ most similar demonstrations will be different. This is a crucial limitation of this approach compared to the others, as it assumes that we are able to acquire labels for any in-context example selected from the pool.

# Active Prompting with Chain-of-Thought for Large Language Models

**Shizhe Diao**[♠], **Pengcheng Wang**[♡], **Yong Lin**[♠], **Rui Pan**[♠], **Xiang Liu**[♣], **Tong Zhang**[♠]
[♠]The Hong Kong University of Science and Technology
{sdiaoaa, ylindf, rpan, tongzhang}@ust.hk
[♡]University of Toronto
pcheng.wang@mail.utoronto.ca
[♣]The University of Hong Kong
xiang.liu@connect.hku.hk

Arxiv'2023, citation 58

# Active Prompt

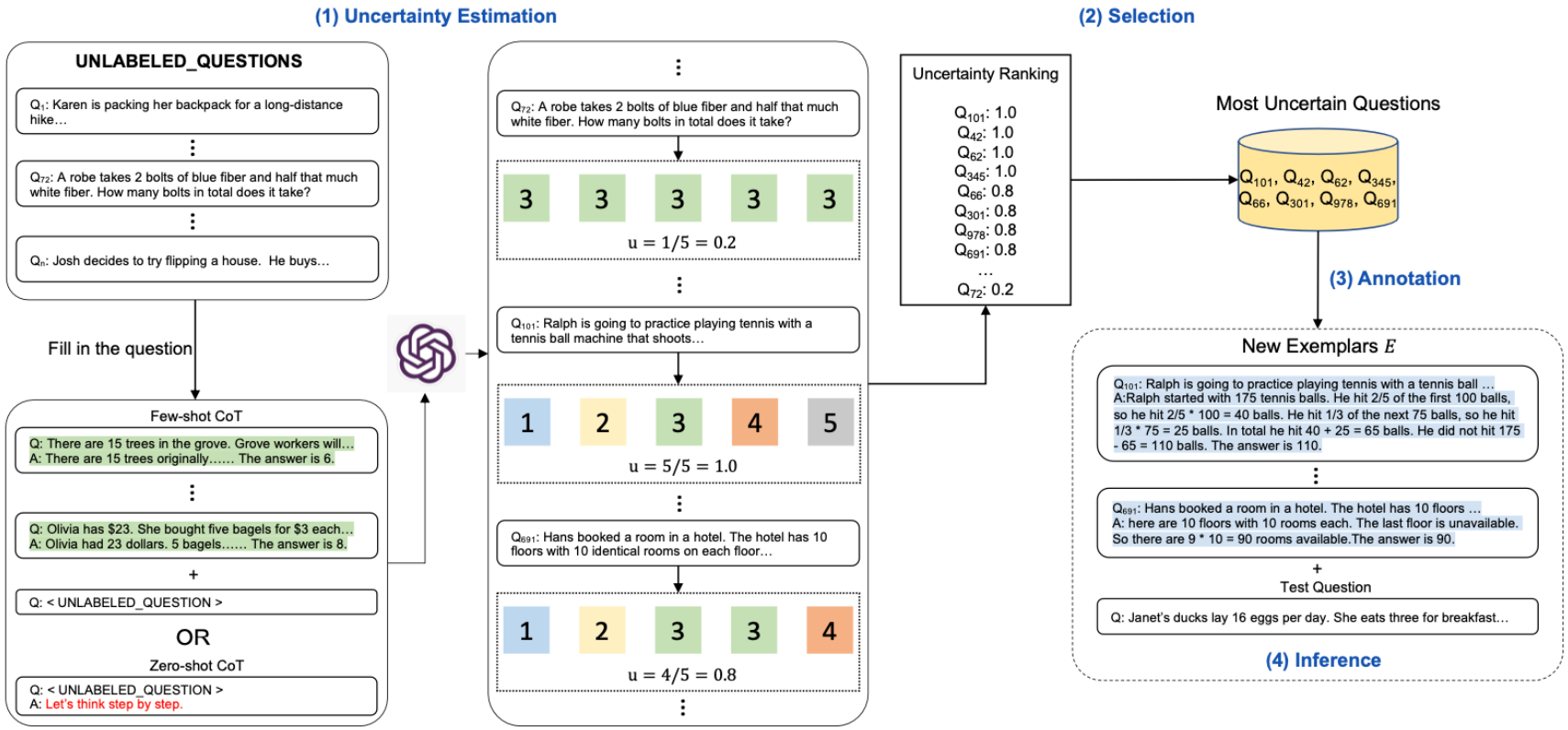



Figure 1: Illustrations of our proposed approach. There are four stages. **(1) Uncertainty Estimation**: with or without a few human-written chain-of-thoughts, we query the large language model $k$ ($k = 5$ in this illustration) times to generate possible answers with intermediate steps for a set of training questions. Then we calculate the uncertainty $u$ based on the $k$ answers via an uncertainty metric (we use disagreement in this illustration). **(2) Selection**: according to the uncertainty, we select the most uncertain questions for annotation. **(3) Annotation**: we involve humans to annotate the selected questions. **(4) Inference**: infer each question with the new annotated exemplars.

# Active Prompt

| METHOD | GSM8K | ASDIV | SVAMP | AQUA | SINGLEEQ | CSQA | STRATEGY | LETTER (4) | AVG. |
|---|---|---|---|---|---|---|---|---|---|
| Prior Best | 55.0[a] | 75.3[b] | 57.4[c] | 37.9[d] | 32.5[e] | 91.2[f] | 73.9[g] | - | - |
| *UL2-20B* | | | | | | | | | |
| CoT | 4.4 | 16.9 | 12.5 | - | - | 51.4 | 53.3 | 0.0 | - |
| SC | 7.3 | 21.5 | 19.4 | 26.9 | - | 55.7 | 54.9 | 0.0 | - |
| *LaMDA-137B* | | | | | | | | | |
| CoT | 14.3 | 46.6 | 37.5 | - | - | 57.9 | 65.4 | 13.5 | - |
| SC | 27.7 | 58.2 | 53.3 | 26.8 | - | 63.1 | 67.8 | 8.2 | - |
| *PaLM 540B* | | | | | | | | | |
| CoT | 56.9 | 73.9 | 79.0 | - | - | 79.9 | 77.8 | 63.0 | - |
| SC | 74.4 | 81.9 | 86.6 | 48.3 | - | 80.7 | 81.6 | 70.8 | - |
| *text-davinci-002* | | | | | | | | | |
| Auto-CoT | 47.9 | - | 69.5 | 36.5 | 87.0 | 74.4 | 65.4 | 59.7 | - |
| CoT | 46.9 | 71.3 | 68.9 | 35.8 | 77.3 | 73.5 | 65.4 | 56.6 | 61.5 |
| SC | 58.2 | 76.9 | 78.2 | 41.8 | 87.2 | 72.9 | 70.7 | 57.6 | 67.9 |
| Random-CoT | 63.9 | 82.3 | 81.1 | 44.1 | 89.4 | 74.5 | 73.3 | 65.5 | 71.8 |
| Active-Prompt (D) | **73.2** | 83.2 | **82.7** | 48.4 | 90.6 | 76.6 | **76.9** | **67.7** | 74.9 |
| Active-Prompt (E) | 71.1 | **83.8** | 81.8 | **50.3** | **93.1** | **78.8** | **76.9** | 66.7 | **75.3** |
| *code-davinci-002* | | | | | | | | | |
| Auto-CoT | 62.8 | - | - | - | - | - | - | - | - |
| CoT | 63.1 | 80.4 | 76.4 | 45.3 | 93.1 | 77.9 | 73.2 | 70.4 | 72.5 |
| SC | 78.0 | 87.8 | 86.8 | 52.0 | 93.7 | 81.5 | 79.8 | 73.4 | 79.1 |
| Random-CoT | 78.6 | 87.1 | 88.0 | 53.1 | 94.0 | 82.1 | 79.4 | 73.3 | 79.4 |
| Active-Prompt (D) | 82.2 | 88.4 | **88.7** | 55.1 | 94.5 | **83.9** | **80.6** | 74.1 | 80.9 |
| Active-Prompt (E) | **83.4** | **89.3** | 87.5 | **57.0** | **95.5** | 82.6 | **80.6** | **76.7** | **81.6** |
| *text-davinci-003* | | | | | | | | | |
| CoT | 61.7 | 78.2 | 77.6 | 46.0 | 91.5 | 76.2 | 72.6 | 70.2 | 71.8 |
| Active-Prompt (D) | **65.6** | **79.8** | **80.5** | **48.0** | **93.1** | **78.9** | **74.2** | **71.2** | **73.9** |

# Active Example Selection for In-Context Learning

**Yiming Zhang** and **Shi Feng** and **Chenhao Tan**
{yimingz0, shif, chenhao}@uchicago.edu
University of Chicago

EMNLP’2023, citation 60

# Active Example Selection By RL

- Maximize the expected accuracy on unseen test examples by getting up to k annotations.

enumerate, so we treat it as a sequential decision making problem: given the pool of unlabeled examples $\mathbf{S}_{\mathcal{X}} = \{x_i\}$, choose one example $x_i$, obtain its groundtruth label $y_i$, append the pair $(x_i, y_i)$ to our prompt, and repeat this process until either the

**Action space and state space.** The action space of the MDP is the set of unlabeled examples plus the special end-of-prompt action: $\mathcal{A} = \mathbf{S}_{\mathcal{X}} \cup \{\perp\}$. After choosing an action $x_i$ we observe its label $y_i$, and the state is defined by the prefix of the prompt $s = (x_1, y_1), (x_2, y_2), \ldots, (x_i, y_i)$.

**Reward.** The reward $r$ can be defined based on an arbitrary scoring function $f$ of the language model LM when conditioned on the prompt $s$, denoted $r = f(\text{LM}_s)$. In practice, we use the accuracy on a labeled validation set as reward.

Large labeled set -> Small labeled set (Follow KATE)

# Large Language Models Are Latent Variable Models: Explaining and Finding Good Demonstrations for In-Context Learning

**Xinyi Wang**[1], **Wanrong Zhu**[1], **Michael Saxon**[1], **Mark Steyvers**[2], **William Yang Wang**[1]
[1]Department of Computer Science, University of California, Santa Barbara
[2]Department of Cognitive Sciences, University of California, Irvine
{xinyi_wang, wanrongzhu, saxon}@ucsb.edu,
msteyver@uci.edu, william@cs.ucsb.edu

NIPS'2023, citation 5

# Method

- The author propose an algorithm to select optimal demonstrations from a set of annotated data with a small LM, and then directly generalize the selected demonstrations to larger LMs.
- Large labeled set -> Small labeled set (Follow KATE)

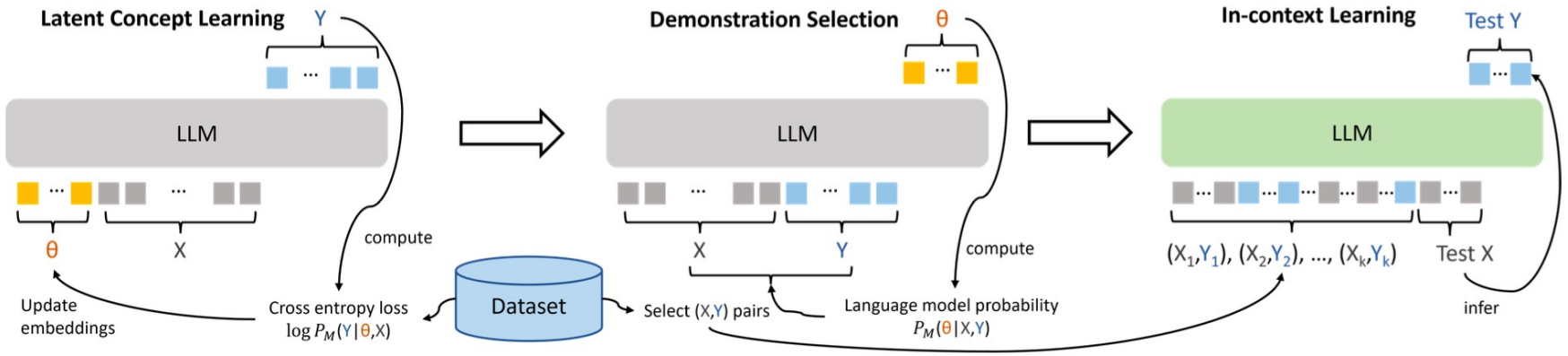



Figure 1: An overview of our proposed two-phased algorithm. Demonstration selection and latent concept learning share the same LLM as demonstration selection needs to reuse the learned concept tokens, while at the in-context learning time, any other generative LLMs can be used. Here we only illustrate the $X \rightarrow Y \leftarrow \boldsymbol{\theta}$ direction. The $Y \rightarrow X \leftarrow \boldsymbol{\theta}$ direction can be illustrated similarly by exchanging $X$ and $Y$ in the above figure.

$$P_M^d(Y|X_1^d, Y_1^d, ..., X_k^d, Y_k^d, X) = \int_\Theta P_M^d(Y|\boldsymbol{\theta}, X) P_M^d(\boldsymbol{\theta}|X_1^d, Y_1^d, ..., X_k^d, Y_k^d, X) d\boldsymbol{\theta} \quad (1)$$

# Method

- Latent Concept Learning (Prompt-tuning)

The fine-tuning objective would then be minimizing $\mathcal{L}(\hat{\theta}^d) = \mathbb{E}_{X,Y}[\ell(X, Y; \hat{\theta}^d)]$, where

$$\ell(X, Y; \hat{\theta}^d) = \begin{cases} -\log P_M^d(Y|\hat{\theta}^d, X) & \text{if } X \to Y \leftarrow \boldsymbol{\theta} \\ -\log P_M^d(X|\hat{\theta}^d, Y) & \text{if } Y \to X \leftarrow \boldsymbol{\theta}. \end{cases}$$

The goal is to learn $\theta^d$ to encode both task $d$ and format information.

- Demonstration Selection

$$\underset{X_1^d, Y_1^d, \ldots, X_k^d, Y_k^d}{\arg\max} \mathbb{E}_X[P_M^d(\theta^d|X_1^d, Y_1^d, \ldots, X_k^d, Y_k^d, X)]$$

As test examples are sampled independent of the demonstrations, and $P_M(X) = P(X)$ according to Assumption 2.1, we have

$$\mathbb{E}_X[P_M^d(\theta^d|X_1^d, Y_1^d, \ldots, X_k^d, Y_k^d, X)] = P_M^d(\theta^d|X_1^d, Y_1^d, \ldots, X_k^d, Y_k^d)$$

If we assume each demonstration is also sampled independently, we have:

$$P_M^d(\theta^d|X_1^d, Y_1^d, \ldots, X_k^d, Y_k^d) = \frac{\prod_{i=1}^k P_M^d(\theta^d|X_i^d, Y_i^d)}{P_M^d(\theta^d)^{k-1}}$$

# Method

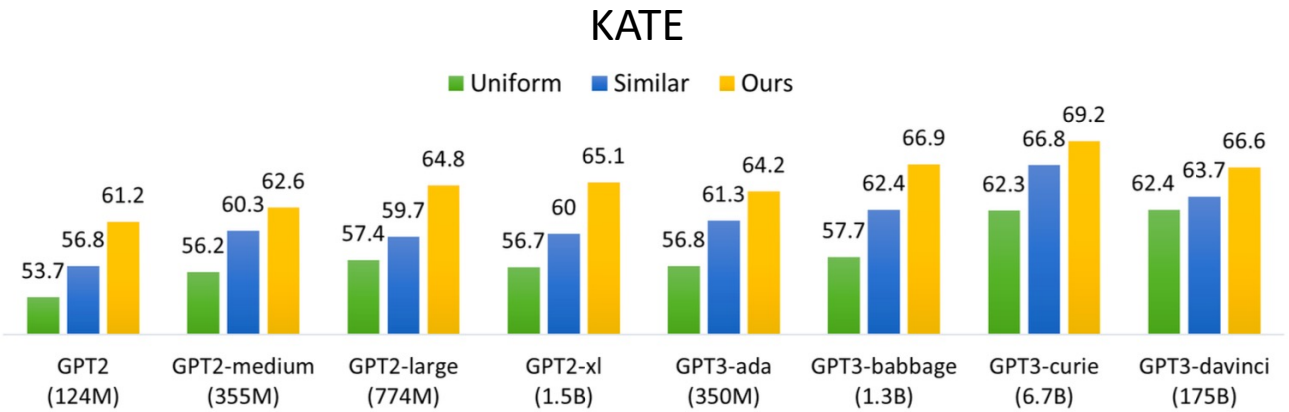



Figure 2: Accuracy of 4-shot in-context learning using demonstrations selected by our method and other baselines, averaged over eight datasets. Our demonstrations are selected using GPT2-large, and the same set of demonstrations is then applied to all other LLMs.

# WHICH EXAMPLES TO ANNOTATE FOR IN-CONTEXT LEARNING? TOWARDS EFFECTIVE AND EFFICIENT SELECTION

**Costas Mavromatis**[†*]
University of Minnesota
mavro016@umn.edu

**Balasubramaniam Srinivasan**[†]
Amazon Web Services
srbalasu@amazon.com

**Zhengyuan Shen**
Amazon Web Services
donshen@amazon.com

**Jiani Zhang**
Amazon Web Services
zhajiani@amazon.com

**Huzefa Rangwala**
Amazon Web Services
rhuzefa@amazon.com

**Christos Faloutsos**
Amazon Web Services
faloutso@amazon.com

**George Karypis**
Amazon Web Services
gkarypis@amazon.com

ICLR'2024, under review, citation 5

# Max Cover

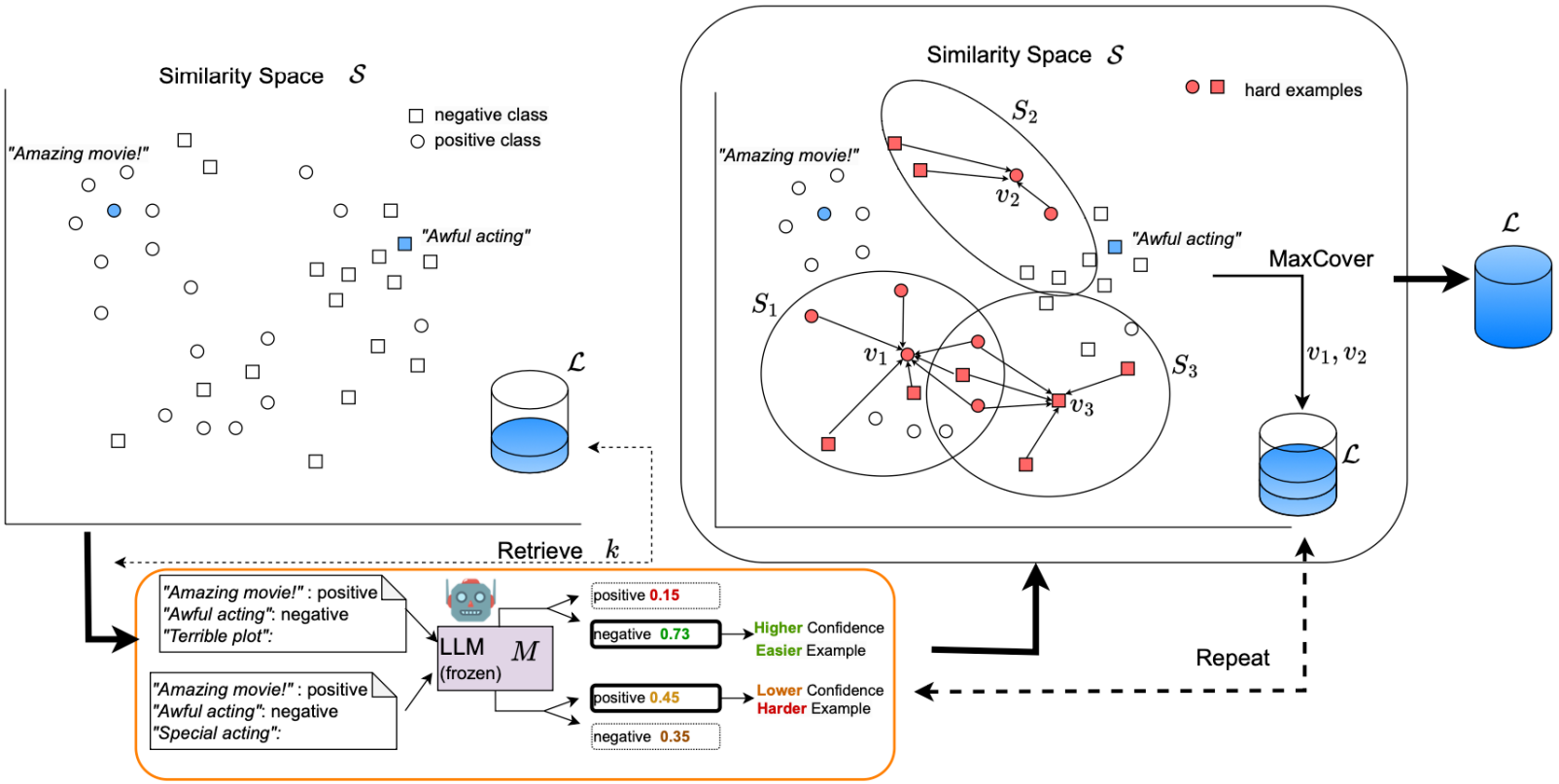



Figure 3: ADAICL algorithm. ADAICL uses $k$-shot ICL to determine which examples the model $M$ is uncertain for (hard examples). Then, it performs diversity-based uncertainty sampling over $\mathcal{S}$ by optimizing the MAXCOVER problem in Equation 3 via Algorithm 1 to identify the examples that help the model learn new information. The process is repeated until the budget $B$ is exhausted, and when done, it returns the annotated set $\mathcal{L}$.

# Max Cover

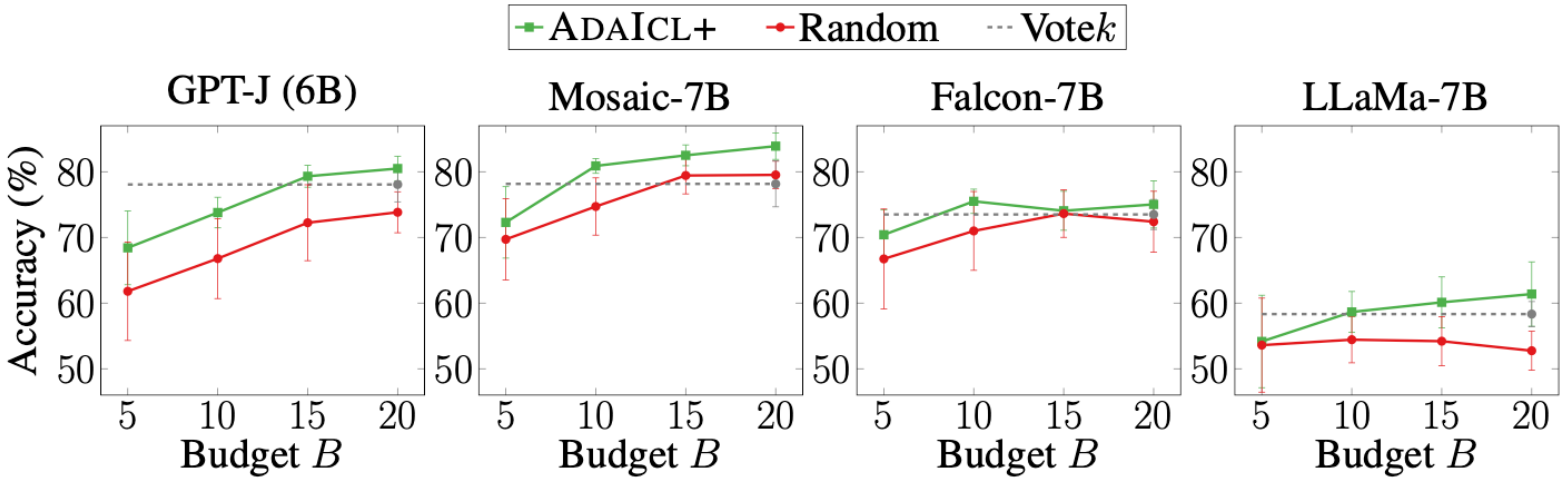
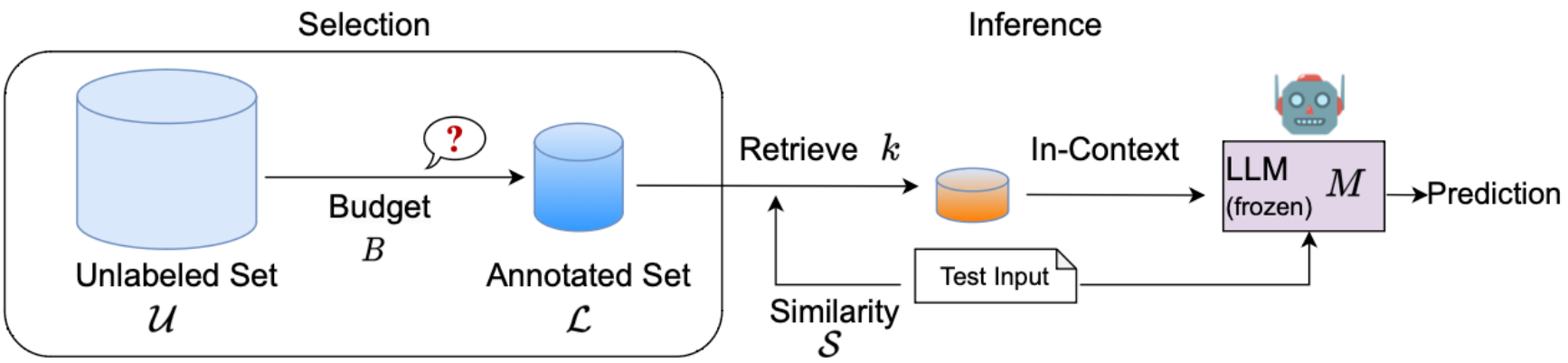



Figure 6: Average results over AGNews, TREC, SST2, and Amazon datasets for four LLMs with similar size.

Table 2: Performance comparison across different retrieval and semantic similarity configurations.

| Retriever, $\mathcal{S}$ → | SBERT-all-mpnet-base-v2 | | | RoBERTa-nli-large-mean-tokens | | | BERT-nli-large-cls-pool | | | Avg. |
|---|---|---|---|---|---|---|---|---|---|---|
| | TREC | SST2 | Amazon | TREC | SST2 | Amazon | TREC | SST2 | Amazon | |
| Pseudo-labeling | $48.56_{\pm 6.33}$ | $69.13_{\pm 3.87}$ | $70.96_{\pm 3.35}$ | $33.98_{\pm 3.68}$ | $74.08_{\pm 4.40}$ | $81.11_{\pm 4.14}$ | $41.27_{\pm 4.24}$ | $77.47_{\pm 1.60}$ | $81.63_{\pm 2.49}$ | 64.24 |
| Random | $54.68_{\pm 1.68}$ | $68.48_{\pm 1.87}$ | $73.95_{\pm 2.03}$ | $37.23_{\pm 2.30}$ | $74.21_{\pm 3.50}$ | $84.46_{\pm 3.21}$ | $34.75_{\pm 2.41}$ | $72.65_{\pm 5.82}$ | $80.20_{\pm 3.34}$ | 64.51 |
| Vote$k$ | $54.81_{\pm 0.49}$ | $73.69_{\pm 9.05}$ | $75.13_{\pm 0.98}$ | $37.77_{\pm 4.65}$ | $76.16_{\pm 2.23}$ | $84.11_{\pm 1.28}$ | $42.43_{\pm 3.34}$ | $80.85_{\pm 2.09}$ | $83.59_{\pm 1.77}$ | 67.61 |
| AdaICL-base | $48.24_{\pm 0.98}$ | $77.86_{\pm 1.02}$ | $75.77_{\pm 3.63}$ | $38.12_{\pm 5.74}$ | $78.12_{\pm 5.30}$ | $\underline{85.93}_{\pm 2.30}$ | $38.15_{\pm 3.10}$ | $78.64_{\pm 2.78}$ | $\underline{85.80}_{\pm 1.75}$ | 67.40 |
| **AdaICL** | $\underline{55.33}_{\pm 2.57}$ | $\underline{79.68}_{\pm 2.47}$ | $\underline{77.73}_{\pm 2.23}$ | $\underline{39.06}_{\pm 3.37}$ | $\underline{81.11}_{\pm 1.50}$ | $85.15_{\pm 0.55}$ | $\underline{44.06}_{\pm 2.49}$ | $\underline{80.85}_{\pm 2.83}$ | $84.65_{\pm 3.52}$ | **69.74** |

# Discussion


Selection
Inference
?
Budget
$B$
Unlabeled Set
$\mathcal{U}$
Annotated Set
$\mathcal{L}$
Retrieve $k$
In-Context
LLM
(frozen)
$M$
Prediction
Test Input
Similarity
$\mathcal{S}$

# Thanks