DM-ICCL: Improving In-Context Learning through DataMap-Based Curriculum

Ido Pinto
ido.pinto@mail.huji.ac.il

Imri Shuval
imri.shuval@mail.huji.ac.il

Michael Finkelson
michael.finkelson@mail.huji.ac.il

Abstract

In this work, we propose a novel approach that combines concepts from Curriculum Learning (CL) and In-Context Learning (ICL) into a new framework we term **D**ata**M**ap **I**n-**C**ontext **C**urriculum **L**earning (**DM-ICCL**). This framework leverages datamaps to categorize training examples into varying difficulty levels (easy, ambiguous, hard) based on confidence and variability measures which are specific to each model-task pair. By integrating similarity-based example selection with these datamaps, we demonstrate that the most effective strategy involves presenting the model with examples of varying difficulties, regardless of the order in which they are presented. This consistency across varying configurations suggests that the primary factor driving performance in our framework is the variance in the difficulty of examples, rather than the difficulty ordering. Our experiments show that the combination of datamaps and similarity consistently achieves the best performance across multiple models and datasets, emphasizing the crucial role of variance in enhancing ICL's effectiveness. Our source code can be found Here.

1 Introduction

In-context learning (ICL) [Brown et al., 2020] [Dong et al., 2023] has gained attention as a powerful method by which large language models (LLMs) perform tasks without updating their internal parameters, relying instead on conditioning through a set of examples presented in the prompt. The selection and ordering of these examples, however, plays a crucial role in determining the effectiveness of the ICL process.

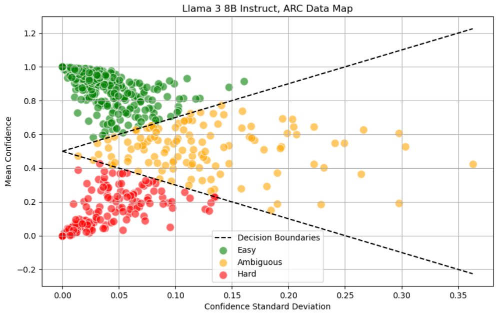

Figure 1: Datamap construction example of Llama3 8B Instruct on the ARC-C dataset. The x-axis represents **variability**, y-axis represents **confidence**, and the colors assigned by difficulty level. Easy examples are characterized by (**High confidence, Low variability**), Hard examples as (**Low confidence, Low variability**), and every point in between categorized as **ambiguous**.

Drawing inspiration from **curriculum learning** (CL) [Bengio et al., 2009][Xu et al., 2020][Feng et al., 2023], a training strategy which posits that models benefit from being exposed to increasingly complex examples over time, recent efforts have sought to apply CL principles within ICL settings [Liu et al., 2023]. Specifically, organizing examples based on difficulty levels, hoping it would allow models to mimic human learning, gradually moving from easier to more challenging tasks. Notably, Swayamdipta et al. [2020] introduced the concept of *datamaps* to analyze the difficulty of samples in datasets, helping in better organization of examples for training.

While these approaches show promise, their efficacy can be significantly amplified when combined with context-based selection mechanisms that account for similarity between the task at hand and the examples in the prompt [Liu et al., 2021], [Su et al., 2022]. In this work, we propose a novel **DM-ICCL** framework that merges the difficulty-based organization from datamaps with similarity-based example selection. We hypothesize that combining datamaps with similarity-driven context selection enhances the model's ability to solve new tasks by presenting it with examples that are both relevant and progressively more challenging.

Our experimental results on the ARC-Challenge and AGNews datasets demonstrate that this combined approach significantly improves performance on Multiple Choice Question Answering (MCQA) tasks. The model achieves the best results when similarity-based examples are carefully integrated into the DM-ICCL framework, confirming that relevance, alongside difficulty, plays a critical role in effective ICL. This combination of similarity and difficulty ordering achieves superior accuracy, highlighting the complementary nature of both strategies.

2 Method

We present a novel example selection strategy for **DM-ICCL** (Datamaps In-Context Curriculum Learning) on **MCQA** (Multiple Choice Question Answering) tasks. Given a LLM $\mathcal{M}_\theta$, MCQA dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ where $y_i \in \{A, B, C, D\}$, and a prompt template $\mathcal{P}$ that accepts k-shot demonstrations followed by a new question x_i.

Consider the following partition of the dataset $\mathcal{D} = (\mathcal{D}_{train}, \mathcal{D}_{eval})$, where $\mathcal{D}_{train}$ acts as our examples pool for few-shot curriculum-based prompting.

The method consists of three main phases:

- Datamap Construction for $\mathcal{M}_\theta$ and $\mathcal{D}_{train}$.
- $\mathcal{D}_{train}$ partitioning to difficulty levels: $\mathcal{D}_{easy}$, $\mathcal{D}_{ambiguous}$, $\mathcal{D}_{hard}$.
- Application using k-shot curriculum-based context on $\mathcal{D}_{eval}$

2.1 Datamap Construction

Our goal is to map each x_i to its mean probability and standard deviation of the correct answer under m evaluations by $\mathcal{M}_\theta$.

Consider a single sample $(x_i, y_i) \in \mathcal{D}_{train}$. We define x_i's **confidence** as the mean probability given to the correct answer y_i by the model across m evaluations, for each of which k examples are randomly sampled for the prompt:

$$\hat{\mu_i} = \frac{1}{m}\sum_{j=1}^{m} p_\theta(y_i|\mathcal{P}(c^jts, c^j_k, x_i))$$

where p_θ denotes the model's output probability with **fixed** parameters θ and $\mathcal{P}(c^j_1, \ldots, c^j_k, x_i)$ denotes the injections of the demonstrations and the new question into the prompt template. We also consider **variability** which measures the standard deviation of the model's probability from the mean across m evaluations:

$$\hat{\sigma_i} = \sqrt{\frac{\sum_{j=1}^{m}\left(p_\theta(y_i|\mathcal{P}(c^j_1, \ldots, c^j_k, x_i)) - \hat{\mu_i}\right)^2}{m}}$$

2.2 Difficulty Assignment

Once we construct our datamap, we can visually observe (Fig. 1) distinct regions of samples for which the model $\mathcal{M}_\theta$ consistently generated the correct (or incorrect) answer with low variability. We partition $\mathcal{D}_{train}$ into $\mathcal{D}_{easy}$, $\mathcal{D}_{ambiguous}$, and $\mathcal{D}_{hard}$ based on a simple heuristic which linearly separates those regions as can be seen in Fig. 1.

2.3 Application

Once the previous step is done, we can apply the method using k_1, k_2, k_3 shots from $\mathcal{D}_{easy}$, $\mathcal{D}_{ambiguous}$, and $\mathcal{D}_{hard}$ as context, respectively.

Our hypothesis suggests that, just as humans benefit from encountering examples of increasing difficulty when facing new challenging problems, the model will similarly benefit. In Section 5 we experiment with different ways to arrange the examples from each difficulty level.

3 Data

- **ARC-Challenge:** a QA dataset of multiple-choice grade-school science questions which is aimed at testing advanced reasoning and knowledge [Clark et al., 2018]. The data set is split into 1119 train samples, 299 validation samples and 1172 samples.
- **AGNews:** a collection of news articles classified into one of four classes (World, Sports, Business, and Sci/Tech) [Zhang et al., 2015]. Due to computational resources limitations, we used just a subsample of the data, matching the ARC-Challange dataset split sizes.

4 Models

The models we use are: Llama3-9b, Llama3-9b-instruct [Llama Team, 2024], Gemma2-9b, Gemma2-9b-instruct [Gemma Team, 2024], and phi3.5 [Microsoft, 2024].

We chose these models as they are all open-source and fit within our resource limitations.

5 Experiments

5.1 Prompt Pre-processing

As pre-processing steps, we convert AGNews into an MCQA format (the ARC dataset already comes in an MCQA format). We standardize the examples such that they contain the question followed by the possible answers in "A., B., C., D." format. We then generate prompts by adding the letter of the correct choice, as can be seen in Section 10.1 and Section 10.2.

To create the datamaps, we used $k = 3$ k-shots and calculated the mean and standard deviation of $num\_evals = 5$ runs with different random seeds.

The confidence and standard deviation were calculated using the softmax over the probabilities of the tokens "A", "B", "C", "D".

We created a separate datamap for each dataset-model configuration over the 2 datasets and the 5 models.

5.2 Context Choices

For each dataset-model configuration, we used the following setups:

- Baseline: 0-shot evaluation.
- Random context: $k = 3, 6$.
- Similarity context: $k = 3, 6$ most similar questions from the training set, using the all-MiniLM-L6-v2 embedder.
- Datamap context: testing these difficulty configurations:
  - **Note:** “Easy,” “Ambiguous,” and “Hard” are abbreviated as E, A, and H.
  - **Difficulty:** {2E, 2A, 2H}, {1E, 1A, 1H}, {3E}, {3A}, {3H}, {6E}, {6A}, {6H}.
  - **Order:** E-A-H, E-H-A, A-E-H, A-H-E, H-E-A, H-A-E.
- Datamap + Similarity context: For each difficulty configuration, selecting the most similar samples within the difficulty levels based on sentence embeddings.

5.3 Metric

To report the accuracy, we check which of the tokens "A", "B", "C", "D" got the highest probability and compare it with the ground truth label.

6 Results

Table 1 summarizes the average accuracy for each configuration across both datasets and models. The baseline 0-shot experiment had the lowest accuracy at 0.80845. The highest accuracy, 0.85299 (a 5.5% improvement), was achieved by combining Datamap and Similarity-based context selection with 2 easy, 2 ambiguous, and 2 hard samples, in that specific order.

A clear pattern emerges: the combination of Datamap and Similarity-based selection with 6 shots—2 easy, 2 ambiguous, and 2 hard examples in different orders—consistently produced the top six results. This indicates that a mix of similar and varied examples is critical to boosting accuracy in ICL.

For the ARC dataset, this configuration consistently ranked in the top five accuracy scores. Likewise, for the AG News dataset, it accounted for 8 of the top 9 scores, with the only exception being the third-highest result achieved by similarity-based selection alone.

Table 2 compares instruction-tuned and non-instruction-tuned models. Instruction-tuned models averaged 0.84933, outperforming non-instruction-tuned models at 0.82769. The combination of Datamap and Similarity-based selection continued to deliver the highest accuracy.

Table 3 highlights that 6-shot configurations slightly outperformed 3-shot ones, with accuracies of 0.84453 and 0.84006, respectively. Both outperformed the 0-shot baseline of 0.80845.

Finally, Figure 2 reinforces these trends with a bar chart showing that the combination of Datamap and Similarity-based selection consistently achieves the highest accuracy across configurations involving easy, ambiguous, and hard examples. This confirms that integrating both strategies leads to better performance.

7 Conclusion

This work introduced a novel approach combining datamaps and similarity-based selection to improve ICL. Our results indicate that the key factor for performance is the variance in difficulty among examples, rather than their specific order. Presenting the model with a diverse set of examples in terms of difficulty consistently led to improved accuracy across datasets and models.

Looking ahead, future work could explore incorporating additional metrics beyond difficulty to better span the space of training examples. This could involve leveraging other characteristics, such as syntactic variance, to ensure the model is exposed to a broader spectrum of examples, further enhancing its generalization and robustness.

8 Limitations

Due to the resource-intensive nature of LLMs, our experiments were limited in their scope:

- We could only test our method on a limited set of models and datasets.
- We used only (relatively) small models and $k = 3, 6$ k-shots.
- Our experiments were limited to the MCQA setting.
- Each experiment was run only once (using a set seed), so our results are prone to variance.

Finally, while the models we used are open-source, the data they were trained on is not, so we do not know whether they've encountered the data we evaluate our method on.

9 Ethical Considerations

In this study, we focus solely on inference using pre-trained large language models (LLMs), which reduces the environmental impact compared to full model training. However, our approach still requires multiple inference runs per sample to compute the data map for each model-dataset pair, which still consumes significant computational resources.

Another point of note is that our DM-ICCL framework uses an existing pool of examples for context without a human-in-the-loop, so it is prone to reinforce the model's answers with harmful or incorrect biases present in the dataset.

References

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 41–48. ACM, 2009.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, Lei Li, and Zhifang Sui. A survey for in-context learning. *ArXiv*, abs/2301.00234, 2023. URL https://api.semanticscholar.org/CorpusID:263886074.

Tao Feng, Zifeng Wang, and Jimeng Sun. Citing: Large language models create curriculum for instruction tuning. *ArXiv*, abs/2310.02527, 2023. URL https://api.semanticscholar.org/CorpusID:263620790.

Google DeepMind Gemma Team. Gemma 2: Improving open language models at a practical size, 2024. URL https://arxiv.org/abs/2408.00118.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. What makes good in-context examples for gpt-3? In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4401–4411, 2021.

Yinpeng Liu, Jiawei Liu, Xiang Shi, Qikai Cheng, and Wei Lu. Let's learn step by step: Enhancing in-context learning ability with curriculum learning. *arXiv preprint arXiv:2302.10738*, 2023.

AI @ Meta Llama Team. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Microsoft. Phi-3 technical report: A highly capable language model locally on your phone, 2024. URL https://arxiv.org/abs/2404.14219.

Hongjin Su, Jungo Kasai, Chen Henry Wu, Weijia Shi, Tianlu Wang, Jiayi Xin, Rui Zhang, Mari Ostendorf, Luke Zettlemoyer, Noah A. Smith, and Tao Yu. Selective annotation makes language models better few-shot learners. *ArXiv*, abs/2209.01975, 2022. URL https://api.semanticscholar.org/CorpusID:252089424.

Swabha Swayamdipta, Roy Schwartz, Nicholas Lourie, Yizhong Wang, Hannaneh Hajishirzi, Noah A. Smith, and Yejin Choi. Dataset cartography: Mapping and diagnosing datasets with training dynamics. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9275–9293. Association for Computational Linguistics, 2020.

Benfeng Xu, Licheng Zhang, Zhendong Mao, Quan Wang, Hongtao Xie, and Yongdong Zhang. Curriculum learning for natural language understanding. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6095–6104, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.542. URL https://aclanthology.org/2020.acl-main.542.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems*, pages 649–657, 2015.

Type	**k-shot**	**Shots per type**	**Ordering**	**Avg Accuracy**	**Avg ARC Accuracy**	**Avg AGNews Accuracy**
Datamap + Similarity	6	[2, 2, 2]	E-A-H	**0.85299**	**0.85410**	**0.85188**
Datamap + Similarity	6	[2, 2, 2]	H-E-A	**0.85290**	**0.85222**	**0.85358**
Datamap + Similarity	6	[2, 2, 2]	E-H-A	**0.85043**	**0.85017**	0.85068
Datamap + Similarity	6	[2, 2, 2]	A-H-E	**0.85009**	0.84625	**0.85392**
Datamap + Similarity	6	[2, 2, 2]	H-A-E	**0.85000**	0.84659	**0.85341**
Datamap + Similarity	6	[2, 2, 2]	A-E-H	**0.84991**	0.84949	0.85034
Datamap + Similarity	3	[1, 1, 1]	E-A-H	0.84940	0.84744	**0.85137**
Similarity	6	-	-	0.84881	0.84403	**0.85358**
Datamap + Similarity	3	[1, 1, 1]	A-H-E	0.84872	0.84608	0.85137
Datamap + Similarity	3	[1, 1, 1]	E-H-A	0.84855	**0.85119**	0.84590
Datamap + Similarity	6	[6, 0, 0]	E-A-H	0.84846	0.84795	0.84898
Datamap + Similarity	3	[1, 1, 1]	H-E-A	0.84744	**0.85137**	0.84352
Datamap + Similarity	3	[1, 1, 1]	H-A-E	0.84718	0.84744	0.84693
Datamap + Similarity	3	[1, 1, 1]	A-E-H	0.84693	0.84710	0.84676
Similarity	3	-	-	0.84445	0.84283	0.84608
Datamap + Similarity	6	[0, 0, 6]	E-A-H	0.84386	0.84488	0.84283
Datamap	6	[6, 0, 0]	E-A-H	0.84343	0.84437	0.84249
Datamap + Similarity	6	[3, 0, 0]	E-A-H	0.84275	0.84198	0.84352
Datamap + Similarity	6	[0, 3, 0]	E-A-H	0.84232	0.84266	0.84198
Datamap + Similarity	6	[0, 6, 0]	E-A-H	0.84172	0.84625	0.83720
Datamap	6	[2, 2, 2]	A-E-H	0.84155	0.84454	0.83857
Datamap	6	[2, 2, 2]	H-E-A	0.84121	0.84727	0.83515
Datamap	6	[2, 2, 2]	H-A-E	0.84053	0.84505	0.83601
Datamap	6	[2, 2, 2]	A-H-E	0.84027	0.84369	0.83686
Datamap	6	[2, 2, 2]	E-H-A	0.84010	0.84505	0.83515
Datamap	6	[0, 0, 6]	E-A-H	0.83985	0.83959	0.84010
Random	6	-	-	0.83951	0.83874	0.84027
Datamap + Similarity	3	[0, 0, 3]	E-A-H	0.83933	0.84044	0.83823
Datamap	6	[2, 2, 2]	E-A-H	0.83814	0.84164	0.83464
Datamap	3	[1, 1, 1]	H-E-A	0.83754	**0.84966**	0.82543
Datamap	3	[1, 1, 1]	H-A-E	0.83695	0.84608	0.82782
Datamap	6	[0, 6, 0]	E-A-H	0.83677	0.83874	0.83481
Datamap	3	[1, 1, 1]	A-E-H	0.83609	0.84522	0.82696
Datamap	3	[1, 1, 1]	A-H-E	0.83609	0.84471	0.82747
Datamap	3	[1, 1, 1]	E-H-A	0.83592	0.84744	0.82440
Datamap	3	[1, 1, 1]	E-A-H	0.83498	0.84522	0.82474
Random	3	-	-	0.83353	0.83737	0.82969
Datamap	6	[0, 3, 0]	E-A-H	0.83294	0.84215	0.82372
Datamap	6	[3, 0, 0]	E-A-H	0.83148	0.83720	0.82577
Datamap	3	[0, 0, 3]	E-A-H	0.82858	0.83106	0.82611
0-shot	0	-	-	0.80845	0.81399	0.80290

Table 1: Experiment data table with Avg, ARC, and AGNews accuracies. Shots per type indicates [#Easy examples, #Ambiguous examples, #Hard examples]. Ordering indicates the order of easy, ambiguous and hard examples. Top 6 results of each column are bolded.

Type	k-shot	Shots per type	Ordering	Avg Instruction-tuned model acc.	Avg Non-instruction-tuned model acc.
Datamap + Similarity	6	[2, 2, 2]	E-A-H	**0.861917**	**0.839590**
Datamap + Similarity	6	[2, 2, 2]	H-E-A	**0.862912**	**0.837884**
Datamap + Similarity	6	[2, 2, 2]	E-H-A	**0.861348**	0.834044
Datamap + Similarity	6	[2, 2, 2]	A-H-E	**0.859642**	0.835751
Datamap + Similarity	6	[2, 2, 2]	H-A-E	**0.860210**	0.834684
Datamap + Similarity	6	[2, 2, 2]	A-E-H	**0.858646**	0.836817
Datamap + Similarity	3	[1, 1, 1]	E-A-H	0.856229	**0.839164**
Similarity	6	-	-	0.856940	0.836604
Datamap + Similarity	3	[1, 1, 1]	A-H-E	0.857793	0.835111
Datamap + Similarity	3	[1, 1, 1]	E-H-A	0.856229	**0.837031**
Datamap + Similarity	6	[6, 0, 0]	E-A-H	0.855375	**0.838097**
Datamap + Similarity	3	[1, 1, 1]	H-E-A	0.855944	0.834684
Datamap + Similarity	3	[1, 1, 1]	H-A-E	0.857082	0.832338
Datamap + Similarity	3	[1, 1, 1]	A-E-H	0.854380	0.835751
Similarity	3	-	-	0.855091	0.828498
Datamap + Similarity	6	[0, 0, 6]	E-A-H	0.848692	**0.836604**
Datamap	6	[6, 0, 0]	E-A-H	0.848549	0.835751
Datamap + Similarity	6	[3, 0, 0]	E-A-H	0.852389	0.828285
Datamap + Similarity	6	[0, 3, 0]	E-A-H	0.853811	0.825085
Datamap + Similarity	6	[0, 6, 0]	E-A-H	0.849687	0.829778
Datamap	6	[2, 2, 2]	A-E-H	0.846559	0.834044
Datamap	6	[2, 2, 2]	H-E-A	0.846274	0.833618
Datamap	6	[2, 2, 2]	H-A-E	0.844852	0.834044
Datamap	6	[2, 2, 2]	A-H-E	0.844852	0.833404
Datamap	6	[2, 2, 2]	E-H-A	0.844425	0.833618
Datamap	6	[0, 0, 6]	E-A-H	0.845421	0.831485
Random	6	-	-	0.843572	0.833404
Datamap + Similarity	3	[0, 0, 3]	E-A-H	0.848976	0.824872
Datamap	6	[2, 2, 2]	E-A-H	0.844852	0.828072
Datamap	3	[1, 1, 1]	H-E-A	0.842292	0.830418
Datamap	3	[1, 1, 1]	H-A-E	0.843003	0.827858
Datamap	6	[0, 6, 0]	E-A-H	0.840159	0.831698
Datamap	3	[1, 1, 1]	A-E-H	0.843857	0.824445
Datamap	3	[1, 1, 1]	A-H-E	0.843003	0.825725
Datamap	3	[1, 1, 1]	E-H-A	0.840728	0.828712
Datamap	3	[1, 1, 1]	E-A-H	0.839733	0.827858
Random	3	-	-	0.842150	0.820606
Datamap	6	[0, 3, 0]	E-A-H	0.842292	0.818899
Datamap	6	[3, 0, 0]	E-A-H	0.838168	0.821459
Datamap	3	[0, 0, 3]	E-A-H	0.835609	0.818046
0-shot	0	-	-	0.844283	0.754693

Table 2: Experiment data table with Avg Instruction-tuned model accuracy and Avg Non-instruction-tuned model accuracy. Shots per type indicates [#Easy examples, #Ambiguous examples, #Hard examples]. Ordering indicates the order of easy, ambiguous, and hard examples. Top 6 results of each column are bolded.

k-shot	Avg Accuracy
0-shot	0.80845
3-shot	0.84006
6-shot	0.84453

Table 3: Average accuracy for different k-shot values.

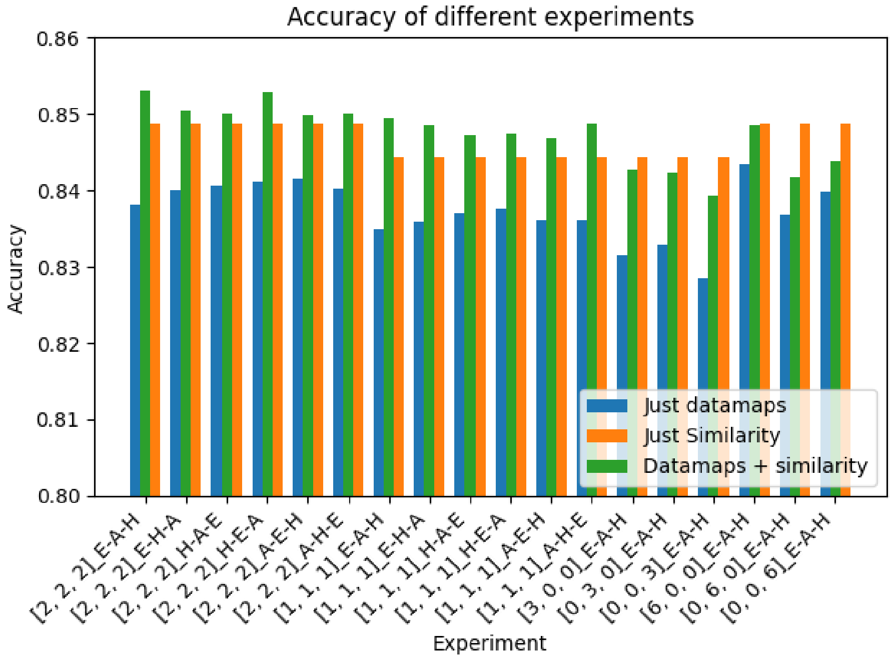

Figure 2: Comparison of just similarity, just datamap and their combinations. The just similarity bars are the same across different orderings as they are only affected by the number of k-shots.

10 Prompts

10.1 ARC prompt

```
Given a question answering task from the 3rd to 9th-grade science exam.
The question contains four options 'A', 'B', 'C' and 'D'.
Select the most appropriate choice that answers the question.

Question: Q1
A. Answer1A
B. Answer1B
C. Answer1C
D. Answer1D
Answer: [Letter of the correct answer]

Question: Q2
A. Answer2A
B. Answer2B
C. Answer2C
D. Answer2D
Answer: [Letter of the correct answer]

Question: Q3
A. Answer3A
B. Answer3B
C. Answer3C
D. Answer3D
Answer: [Letter of the correct answer]

Question: Actual Question
A. Answer4A
B. Answer4B
C. Answer4C
D. Answer4D
Answer:
```

10.2 AG News prompt

```
Classify the news articles into the categories.
The categories are labeled 'A', 'B', 'C' and 'D'.
Select the most appropriate category for the news article.

Question: Q1
A. World
B. Sports
C. Business
D. Sci/Tech
Answer: [Letter of the correct answer]

Question: Q2
A. World
B. Sports
C. Business
D. Sci/Tech
```

```
Answer: [Letter of the correct answer]

Question: Q3
A. World
B. Sports
C. Business
D. Sci/Tech
Answer: [Letter of the correct answer]

Question: Actual Question
A. World
B. Sports
C. Business
D. Sci/Tech
Answer:
```

10.3 Example for a full prompt of AG News

```
Classify the news articles into the categories.The categories are labeled 'A', 'B', 'C' and 'D'.
Select the most appropriate category for the news article.
News: Several Mass. Communities Eye Wind Power (AP) AP - Wind power projects are in various stages
around the state, from the Atlantic coast in the east to the wooded slopes of the Berkshire Mountains
in the west, gigantic towers whose hurtling blades are designed to create clean energy.
A. World
B. Sports
C. Business
D. Sci/Tech
Answer: D

News: Auto Makers Sue to Block New California Emission Rule DALLAS -- Auto makers filed suit Tuesday
to block California from forcing the companies to make cars and trucks that emit lower levels of
global-warming gases.
A. World
B. Sports
C. Business
D. Sci/Tech
Answer: C

News: WTO Rules Against EU Protection of Goods (AP) AP - The United States and Australia prevailed in
an interim ruling by the World Trade Organization in a dispute over protection given by the European
Union to its regional goods such as Champagne wine and Feta cheese, trade officials said Thursday.
A. World
B. Sports
C. Business
D. Sci/Tech
Answer: A

News: Calif. Aims to Limit Farm-Related Smog (AP) AP - Southern California's smog-fighting agency
went after emissions of the bovine variety Friday, adopting the nation's first rules to reduce
air pollution from dairy cow manure.
A. World
B. Sports
```

C. Business
D. Sci/Tech
Answer: