DiffTune: A Diffusion-Based Approach to Diverse Instruction-Tuning Data Generation

Abstract

Instruction tuning has become pivotal in enhancing the adaptability and responsiveness of Large Language Models (LLMs) to human instructions. Despite its critical role, current methods for generating instruction-tuning datasets exhibit significant bottlenecks, primarily in terms of high cost and limited diversity. However, as previously shown in the literature, the diversity of an instruction-tuning dataset is crucial to LLM’s downstream performance. To address these challenges, we propose a Diffusion Language Model (DiffLM)-based technique to generate unlimited diverse instructions at a low cost. Specifically, we have enhanced the variability of instructions by strategically modifying the sampling process within the DiffLM. Our method presents the opportunity to augment any existing instruction-tuning dataset, thereby enriching its content and potential utility. Both automatic and human evaluation show that our generated instructions achieve high quality and better n-gram diversity than the original dataset. Instruction tuning of LLaMA on the augmented dataset delivers better instruction following capability and superior performance on a broad set of benchmarks, indicating the effectiveness of our instruction generation method.

1 Introduction

Large Language Models (LLMs), particularly following the advent of ChatGPT, have seen a surge in popularity due to their impressive performance capabilities [1][2]. To maximize LLMs’ potential and adapt pre-trained models to specific domains or downstream tasks, instruction tuning emerges as an indispensable step [3][4]. It involves the generation of bespoke datasets that guide Large Language Models (LLMs) to respond more effectively to human instructions across varying tasks and domains.

Existing instruction-tuning techniques generally fall into two categories: human-labeled and machine-generated approaches. Human-labeled methods [3][5] are highly accurate and contextually rich, but is difficult to scale up and expensive to procure. Machine-generated techniques Wang et al. [6], Peng et al. [7], Honovich et al. [8] are easily scalable but lack the necessary diversity, creating a gap between the instructions in the dataset and real-world user prompts. Also, generating datasets by querying commercial language models also involves costs, which could be substantial [9][7]. The inherent limitations of existing instruction-tuning techniques underscore the need for a more effective, scalable, and cost-efficient approach, forming our research’s central theme.

Given the aforementioned challenges, we propose DiffTune, a novel data generation technique utilizing the Diffusion Language Model (DiffLM). The diffusion model, as a kind of generative model, works by simulating a process of random walks from a simple initial distribution toward the target distribution, resulting in nuanced and detailed data generation. Building upon this foundation, DiffTune innovatively leverages the inherent properties of a Diffusion Language Model (DiffLM). By replacing the original sampling strategy within the DiffLM with our topic diversity enhancing
sampling, DiffTune is imposed to generate high-quality instruction-tuning datasets with enhanced diversity at a lower cost.

We put our novel DiffTune-generated dataset to the test by finetuning an accessible LLM, LLaMA [10]. Our methodology involved augmenting or replacing the original datasets with the instructions generated by DiffTune. The LLaMA model, when finetuned with our DiffTune-generated dataset, displayed a remarkable increase in its instruction-following capabilities in terms of higher validity and human preference. This underscores the potential of DiffTune to optimize LLM performance cost-effectively.

2 Data Collection for DIFFIT

This section introduces the fully automatic collection process of our instruction tuning dataset DIFFIT. The overall data collection process is shown in Figure 1.

2.1 Diffusion Language Model Training

Given an existing instruction-tuning dataset \( \{(X^e_t, Y^e_t)\} \), where \( X^e_t \) and \( Y^e_t \) are the instruction and output of an instance in the dataset, we train a DiffLM on its instruction set \( \{X_t\} \) as described previously. After this stage, we obtain a trained DiffLM, denote it \( M_{\text{Diff}} \). We can now sample from it:

1) sample a length from the empirical length distribution of the existing instruction set \( l_i \sim L(\{X_t\}) \), and
2) sample a noise \( Z_T \in \mathbb{R}^{l_i \times d} \sim N(0, I) \). The generation \( M_{\text{Diff}}(Z_T) \) is a sequence in the same domain of the original instruction set \( \{X_t\} \).

2.2 Diffusion Language Model Sampling

We sample from the trained \( M_{\text{Diff}} \) to generate new, diverse, and high-quality instructions. Lovelace et al. [11] demonstrated that DiffLM could generate diversified text sequences with low memorization of its training set when using a noise sampled from \( N(0, I) \). In our method, we further increase the diversity of our sampled instruction set by adopting an innovative sampling strategy to cover the rare topics, concepts, and formats mapped to the long tail of the sampling noise distribution.

Inspired by the in-breadth evolving strategy mentioned in Xu et al. [12], we propose the topic diversity enhancing sampling strategy. After sampling the noise from a standard Gaussian \( Z_T \in \mathbb{R}^{l_i \times d} \sim N(0, I) \), we randomly select 30% of the tokens and sample them from a distribution of a much higher variance \( N(0, 10I) \). This strategy resembles the process of randomly inserting rare tokens into the sequence. With the remaining 70% originally sampled tokens to control the overall format and \( M_{\text{Diff}} \)'s powerful BART decoder, the generation’s quality is only slightly compromised. The post-processing step can mitigate the slightly lower generation quality.

Figure 1: The dataset collection process of DiffTune.
<table>
<thead>
<tr>
<th>Dataset</th>
<th>Size</th>
<th>Avg. len.</th>
<th>#Unique Tokens</th>
<th>4-gram Rep.↑</th>
<th>4-gram SelfBLEU↓</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unnat. Inst.</td>
<td>68478</td>
<td>93.92</td>
<td>42455</td>
<td>0.61</td>
<td>0.85</td>
</tr>
<tr>
<td>Self-Instruct</td>
<td>82439</td>
<td>35.59</td>
<td>19920</td>
<td>0.70</td>
<td>0.90</td>
</tr>
<tr>
<td>Alpaca</td>
<td>52002</td>
<td>22.79</td>
<td>21027</td>
<td>0.43</td>
<td>0.69</td>
</tr>
<tr>
<td>GPT4-Alpaca</td>
<td>52002</td>
<td>22.79</td>
<td>21027</td>
<td>0.43</td>
<td>0.69</td>
</tr>
<tr>
<td>Code-Alpaca</td>
<td>20022</td>
<td>29.90</td>
<td>8671</td>
<td>0.55</td>
<td>0.79</td>
</tr>
<tr>
<td>OASST1</td>
<td>55668</td>
<td>25.09</td>
<td>36150</td>
<td>0.83</td>
<td>0.96</td>
</tr>
<tr>
<td>S.A.D.</td>
<td>52000</td>
<td>30.24</td>
<td>55455</td>
<td>0.19</td>
<td>0.51</td>
</tr>
<tr>
<td>DIFFIT</td>
<td>52000</td>
<td>28.64</td>
<td>49322</td>
<td>0.14</td>
<td>0.49</td>
</tr>
</tbody>
</table>

Table 1: Evaluation of existing similarly-sized datasets’ input instructions in terms of different diversity metrics. S.A.D.: 52K instructions sampled from the concatenation of ShareGPT + Alpaca + Dolly’s instruction set with stratified sampling. ↑: Higher is better. ↓: Lower is better. The best and second-best results are labeled in **bold** and *underline*, respectively.

## 2.3 Instruction Post-Processing

In this step, we filter out the sampled instructions from $M_{Diff}$ with a perplexity threshold. We use GPT2-Large to compute the perplexity. This simple yet effective post-processing strategy can drastically decrease the average perplexity of the generated dataset by four times. However, since perplexity computation largely depends on the evaluation model, which is not explicitly pre-trained on the instruction domain, this process potentially filters out some valid but highly diversified instructions. We leave developing a better post-processing strategy for this process as future work.

## 2.4 Instruction Output Generation

After obtaining a predefined number of valid instructions with the previous steps, we generate the output by prompting an existing LLM. We iterate over all generated instructions, prompt the LM with the instruction, and collect the LLM’s response. We filter out the instructions with an invalid response (e.g., the LLM’s output contains no helpful information or deems the instruction as a not self-contained sequence). The remaining instruction-output pairs form our instruction-tuning dataset.

## 3 Instruction Data Analysis

We apply the method introduced in Section 2 to the concatenation of three open-source datasets’ instruction sets: ShareGPT[^1], Dolly [5] and Alpaca [13]. The combined dataset ShareGPT-Alpaca-Dolly (S.A.D.) contains 107,442 instructions. We sample 1000 instructions from it with stratified sampling as the test set, while the remaining 106,442 instructions are used for training DiffLM.

We sampled an instruction set with a DiffLM trained on the joint S.A.D.’s training set, using a BART-Large as its decoder. The output for each instruction was collected with gpt-3.5-turbo. The resulting dataset contains 52,000 diverse instructions with high-quality outputs. We name our dataset DIFFIT (Diffusion-based Instruction Tuning dataset).

We compare our dataset DIFFIT with several open-sourced instruction tuning datasets in Table 1 in terms of instruction diversity. Among the similar-sized datasets, S.A.D. and our DIFFIT achieves the highest unique token counts. Although DIFFIT has lower unique token counts compare to S.A.D., it achieves a better $n$-gram diversity in terms of 4-gram Repetition and 4-gram SelfBLEU. This suggests that compared to DiffLM’s training instruction set, the sampled instructions from DiffLM can cover more new complex concepts or phrases.

## 4 Instruction Tuning Experiments

We conduct instruction-tuning on a pre-trained LLM, LLaMA [10] with our sampled DIFFIT dataset. In this section, we compare a LLaMA 7B fine-tuned on DIFFIT with LLaMA 7B finetuned on similar-sized instruction-tuning datasets with the same training settings.

[^1]: We use an open-source version of ShareGPT from https://hf.co/datasets/anon8231489123/ShareGPT_Vicuna_unfiltered
<table>
<thead>
<tr>
<th>Model &amp; Dataset</th>
<th>MMLU</th>
<th>GSM</th>
<th>Codex-HumanEval</th>
<th>TydiQA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>Avg.</strong></td>
<td>0-shot</td>
<td>5-shot</td>
<td>CoT</td>
</tr>
<tr>
<td>LLaMA 7B [10]</td>
<td>31.9</td>
<td>35.2</td>
<td>6.0</td>
<td>9.0</td>
</tr>
<tr>
<td>+ Unnat. Inst. [8]</td>
<td>42.9</td>
<td>38.1</td>
<td>3.5</td>
<td>5.0</td>
</tr>
<tr>
<td>+ Self-Instruct [6]</td>
<td>35.7</td>
<td>33.2</td>
<td>4.0</td>
<td>6.5</td>
</tr>
<tr>
<td>+ Alpaca [13]</td>
<td>41.5</td>
<td>40.3</td>
<td>7.0</td>
<td>10.0</td>
</tr>
<tr>
<td>+ GPT4-Alpaca [7]</td>
<td>42.6</td>
<td>38.3</td>
<td>6.5</td>
<td>10.0</td>
</tr>
<tr>
<td>+ Code-Alpaca [14]</td>
<td>34.7</td>
<td>34.5</td>
<td>6.5</td>
<td>10.0</td>
</tr>
<tr>
<td>+ OASST1 [15]</td>
<td>32.9</td>
<td>29.7</td>
<td>6.0</td>
<td>6.5</td>
</tr>
<tr>
<td>+ S.A.D.</td>
<td>37.4</td>
<td>27.3</td>
<td>5.5</td>
<td>14.0</td>
</tr>
<tr>
<td>+ DIFFIT</td>
<td>39.7</td>
<td>33.6</td>
<td>7.5</td>
<td>14.5</td>
</tr>
<tr>
<td>+ S.A.D. (2x Training)</td>
<td>38.7</td>
<td>30.5</td>
<td>4.5</td>
<td>13.5</td>
</tr>
<tr>
<td>+ S.A.D.+DIFFIT</td>
<td>40.6</td>
<td>32.7</td>
<td>6.5</td>
<td>14.5</td>
</tr>
</tbody>
</table>

The results on a suite of automatic evaluations on LLM’s general capability are shown in Table 2. Instruction tuning on DIFFIT achieves better performance than instruction-tuning on similarly-sized DiffLM training set S.A.D. and augmenting S.A.D. with the generated dataset can further improve LLM’s general performance. This suggests that DiffTune can generate high-quality instructions that can augment existing diverse instruction-tuning datasets to increase LLM’s general capabilities.

We show the automatic evaluation on AlpacaFarm in Figure 2. We find that although a mixture of 52K S.A.D. instructions is itself a diverse dataset, DIFFIT sampled by DiffTune further increases the response quality by achieving a higher win rate against text-davinci-003. S.A.D. augmented by DIFFIT achieves a 2.2% win rate increase compared to LLaMA trained on S.A.D. with the same training steps. This shows the effectiveness of DiffTune as both an instruction-tuning generation method and an instruction set augmentation approach.

Lastly, we conduct a human evaluation of the validity and helpfulness of LLM’s response to real-world human instructions. In Figure 3, we illustrate the percentage of valid responses evaluated by human evaluators. The original 52K S.A.D. dataset achieves better response validity when augmented with DIFFIT, achieving an increase of 1.3 percentage points brought by our method.

5 Conclusion

We introduce DiffTune, a novel method for generating instruction-tuning datasets that overcome the limitations of current techniques. By leveraging the capabilities of Diffusion Language Models and revising the sampling strategy, DiffTune generates diverse, high-quality instruction-tuning datasets in a cost-effective manner. The superior performance of the LLaMA model, when finetuned with our DiffTune-generated dataset, emphasizes the efficacy of our approach. Both automatic and human evaluations underscore the quality and diversity of the data generated by DiffTune, showcasing its potential to optimize the performance of Large Language Models across varied tasks and domains.
References


[10] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Thimothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez,


A Related Works

Instruction-Tuning Datasets Recently, instruction tuning on LLMs has been a hot research area in NLP [6, 4, 17, 18]. The dataset used for LLM instruction tuning has to be diverse to cover various tasks, scenarios, and input formats. To guarantee the diversity requirement of the dataset, previous literature generates their instruction data from a variety of existing NLP datasets [3, 4, 17, 19, 20], various online forums [21], human labelling [5], crowdsourcing [15, 22], and generation by existing (proprietary) LLMs [6, 13, 7, 8, 12, 23]. However, LLM-generated instruction tuning datasets frequently suffer from lower diversity and less authenticity than human-generated datasets, and human-labeled datasets have a high cost for dataset generation. Although crowd-sourcing-based instruction tuning datasets achieve better variety and can best reflect real-world user prompts, collecting a large set of user inputs is still costly. This paper introduces a cost-effective method of instruction-tuning dataset generation that can construct or augment an existing dataset with crowdsourcing-level quality and diversity with no extra human labeling.

Diffusion Model for Language Generation Different from the mainstream auto-regressive language models (ARLM) [1, 10] which generates texts token by token, diffusion language models (DiffLM) fall into the category of non-auto-regressive language models (NARLM), which generate all tokens in parallel. Diffusion LM was first introduced by Li et al. [24], where the authors trained a diffusion model in the token embedding space. DiffLM has been applied to controllable text generation [24, 25], unconditional text generation [26, 28] and sequence-to-sequence tasks [29, 32]. On the task of unconditional generation, compared to ARLM, DiffLM can achieve more robust and efficient text sequence generation [33] and higher generation diversity [11]. This paper adopts the DiffLM proposed by Lovelace et al. [11] for diverse and high-quality instruction generation.

B Backgrounds on Diffusion Language Models

Diffusion models [34, 35] aim to approximate a target distribution \( p(x) \) by learning a reversible transition between it and a Gaussian distribution. The forward process takes a sample from the target distribution \( \{z_0, z_1, \ldots, z_T\} \), where \( q(z_{t+1} | z_t) = \mathcal{N}(\sqrt{1 - \beta_t} z_t, \beta_t I) \) with some variances \( \beta_t \). The inversion of the forward process is called denoising, where one samples from Gaussian distribution \( z_T \sim \mathcal{N}(0, I) \) and sequentially produces a chain of less noisy samples \( \{z_T, z_{T-1}, \ldots, z_0\} \), where the final element \( z_0 \) is a sample from the target distribution \( p(x) \). For that, one trains the denoising neural network \( f_\theta(z_t, t) \), which approximately recovers the original sample from target distribution, \( x \), from its noisy version \( z_t \). Specifically, for any \( x \sim p(x) \) and any time step \( t \), one generates the noisy sample \( q(z_t | x) \) with a forward process, and recovers it so that \( x_\theta \approx x \). When the denoising network is trained, one can generate samples from the target distribution using the denoising Markov chain described above. Going from \( z_t \) to \( z_{t-1} \) requires sampling from the distribution \( p(z_{t-1} | z_t) := \mathcal{N}(\mu_t(z_t, x_\theta), \sigma^2_t I) \), where \( \mu_t(z_t, x) \) has a closed form solution.

Applying diffusion models to NLP is not straightforward because of the discrete nature of language. We use the model suggested by Lovelace et al. [11], where they modeled the latent space of the encoder-decoder language model with diffusion. In particular, they used BART [36], because it was trained with the denoising objective on the latent representation. Hence, the approximate samples from the diffusion model would be meaningfully decoded. Instead of learning to generate a sample from the latent space, one needs to sample a sequence of vectors, which would be ideally decoded into a valid sentence. For that, given a length \( l \) and a sampled noise \( Z_T \in \mathbb{R}^{l \times d} \sim \mathcal{N}(0, I) \), the denoising \( X_T \) is a sequence in the target domain.

C Instruction Data Generation Analysis

We apply DiffTUNE to different existing instruction tuning datasets, different DiffLM decoder size and different sampling strategies. We analyze these three aspects of instruction generation settings one by one in the following subsections.

9
Table 3: The comparison of sample diversity and quality among different DiffLM decoder selection and DiffLM sampling strategies based on 1000 sampled instructions. ↑: Higher is better. ↓: Lower is better. The setting with the highest performance for each metric is labeled in bold, while the second-highest is labeled in underline.

C.1 Experiment Details

To maximize the diversity of the sampled new instructions from DiffLM, we train our DiffLM on the concatenation of the following open-source datasets’ instruction set: 1) ShareGPT. It is collected from publically-shared real-world user dialogues with ChatGPT. We filtered the dataset only to contain first-round user inputs with a valid response from ChatGPT. The filtered dataset contains 40,428 instructions. 2) Dolly. It contains 15,011 instructions created by Databricks employees. 3) Stanford Alpaca. It is generated with an modified self-instruct strategy [6] using text-davinci-003, containing 52,002 instructions.

The combined dataset ShareGPT-Alpaca-Dolly (S.A.D.) contains 107,442 instructions. We sample 1000 instructions from it with stratified sampling as the test set, while the remaining 106,442 instructions are used for training our DiffLM.

In this section, we train two DiffLMs with different DiffLM decoder sizes: BART-Base or BART-Large. After training, we test four different sampling strategies:

1. **Standard Gaussian.** We sample from the DiffLM with \( Z_T \in \mathbb{R}_{l_i \times d} \sim \mathcal{N}(0, I) \).
2. **Student T.** We sample from the DiffLM with \( Z_T \in \mathbb{R}_{l_i \times d} \sim t_2 \). The noise distribution puts a higher probability on the long tail compared to the original strategy.
3. **30% Higher Variance.** We apply the sampling strategy introduced in **DIFFTUNE**.
4. **100% Higher Variance.** We sample from a Gaussian with a higher variance for all tokens: \( Z_T \in \mathbb{R}_{l_i \times d} \sim \mathcal{N}(0, 10I) \).

We sample 1000 instructions from each configuration with beam search. The sampled instruction set will be evaluated with the following metrics:

1. **Repetition** [37] measures generation diversity by the proportion of repetitive n-grams:
   \[ rep_n = \left( 1 - \frac{\text{unique} - \text{n-grams}(z)}{\text{total} - \text{n-grams}(z)} \right) \].
2. **n-gram Diversity** [37] measures generation diversity by considering different n-gram repetitions:
   \[ \text{div} = \prod_{n=2}^{4} \left( 1 - \frac{\text{rep}_n}{100} \right) \].
3. **SelfBLEU** [38] measures generation diversity by computing the average of each generated instance’s BLEU score against all others.
4. **Memorization** [11] measures generation diversity by computing the proportion of 4-grams from the generated sequences that exist in the training set.
5. **MAUVE** [39] measures generation quality by considering the token distribution closeness between the generated and reference sets. We compute the MAUVE score against S.A.D.’s test set.
6. **Perplexity (ppl)** measures generation quality by how likely a language model can generate the sequence. We compute ppl with GPT2-Large.
C.2 Comparisons of Training Settings

We sample 1000 instructions from each setting with an applied perplexity threshold of 150, which aligns with our final instruction generation process. We compare the diversity and quality of the generated instruction set with S.A.D.’s test set. The results are shown in Table 3.

**DiffLM’s decoder model size.** When the instructions are sampled from standard Gaussian, using a smaller BART-Base as DiffLM’s decoder achieves slightly on-par or better diversity and quality across all metrics except for perplexity, which aligns with the observation from Lovelace et al. [11]. However, when using a different sampling strategy, using a larger decoder illustrates a different trend. For generation quality, when using the same sampling strategy, BART-Large settings always achieve a lower perplexity compared to its BART-Base counterpart. The higher generation quality from a BART-Large decoder is also observed during our case studies. For generation diversity, when using a sampling distribution distant from a standard Gaussian, using BART-Large generally achieves a higher diversity gain.

**DiffLM’s sampling strategy.** Although all tested settings achieve a higher diversity across all metrics compared to S.A.D.’s test set, we found that using a noise distribution other than standard Gaussian always achieves a more diversified instruction generation. Although sometimes the generated instructions include grammatical errors or unknown concepts, they can be denoised in the output generation process by larger LLMs and can better resemble real-world user inputs, where the prompts are not always perfect.

We provide a simple case study of using different percentages of high-variance noises in Table 4. We begin with a noise matrix $Z_T \sim \mathcal{N}(0, I)$, and gradually substitute a specific percentage of its column vectors with sampled vectors from $\mathcal{N}(0, I)$ and observe the decoded output. We keep sampling until the generation achieves a perplexity below 150, which takes around 2 rounds of generation.

We observed that the generated sequences are all in the format of instructions, while a higher percentage of vectors sampled with higher variance is more likely to introduce grammatical errors (e.g., “i am”) or unknown concepts (e.g., “Iafkenhoek”). This phenomenon resembles real-world user inputs since similar grammatical errors and typos are common in ShareGPT’s instruction set. In our dataset generation process, we adopt the setting of using a BART-Large decoder and a 30% Higher Var sampling strategy.

<table>
<thead>
<tr>
<th>% Tokens Sampled With Higher Var.</th>
<th>DiffLM Generation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0%</td>
<td>Could you list few 10 most important things to prepare for the entrance examination? Think about the factors in order to determine your aptitude. Please write in English language.</td>
</tr>
<tr>
<td>10%</td>
<td>I am an employee at a large endo wrapping company in Hengshui, China. Give me some suggestions for a new cover letter and resume. I would love to have a good job description.</td>
</tr>
<tr>
<td>30%</td>
<td>Make me a list of things for the course I should do and be prepared. I’m doing a user design course, I am preparing for a class, but I don’t know what to do about it.</td>
</tr>
<tr>
<td>50%</td>
<td>Write a poem in the style of Iafkenhoek explaining how humans will overcome a number of factors of mental and emotional goals, which may or may not be attainable. Create a short film about your dreams for humanity.</td>
</tr>
</tbody>
</table>

Table 4: A sample from DiffLM trained on S.A.D.’s instruction set with a BART-Large decoder. We begin with a noise matrix $Z_T \sim \mathcal{N}(0, I)$, and gradually substitute column vectors of $Z_T$ with vectors sampled from $\mathcal{N}(0, 10I)$ and observe the corresponding changes for its decoded generation.
### Statistics

<table>
<thead>
<tr>
<th>Statistics</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td># of data</td>
<td>52000</td>
</tr>
<tr>
<td># of unique input tokens</td>
<td>49322</td>
</tr>
<tr>
<td># of unique output tokens</td>
<td>74079</td>
</tr>
<tr>
<td>Avg. input length (in words)</td>
<td>24.4</td>
</tr>
<tr>
<td>Avg. output length (in words)</td>
<td>80.7</td>
</tr>
</tbody>
</table>

Table 5: Basic statistics of the DIFFIT dataset.

**Figure 4:** The most common root verb-noun combinations in DIFFIT’s instruction set. The inner circle illustrates the root verbs, while the outer circle illustrates the corresponding direct nouns.

### D Statistics of DIFFIT

#### D.1 Basic Statistics

We include the basic statistics of DIFFIT in Table 5.

#### D.2 Verb-Noun Analysis

Following previous practices of instruction diversity analysis [6, 7], we analyze the most common verb-noun combinations in the sampled instructions. We extract the root verb and their corresponding direct-object noun of each instruction and plot the verb-noun combinations with a frequency higher than 10 in Figure 4. We observe a large variety in the verbs used in the dataset, with the instructions covering different generation types, including story, code, table, list, etc. It is also worth noticing that the verb “use” appears frequently in the instructions, which is usually intended to add specifications to the task scenario (e.g., use a specific library or use a particular tool), which resemble typical real-world user inputs to LLMs.
### Table 6: Dataset construction cost of several existing open-source instruction-tuning datasets with similar sizes.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Size</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Self-Instruct</td>
<td>82K</td>
<td>$600</td>
</tr>
<tr>
<td>Unnatural Instructions</td>
<td>68K</td>
<td>$1370</td>
</tr>
<tr>
<td>Alpaca</td>
<td>52K</td>
<td>$500</td>
</tr>
<tr>
<td>GPT4-Alpaca</td>
<td>52K</td>
<td>$456</td>
</tr>
<tr>
<td>DIFFIT</td>
<td>52K</td>
<td>$27.8</td>
</tr>
</tbody>
</table>

Table 6: Dataset construction cost of several existing open-source instruction-tuning datasets with similar sizes.

![Preference Table](chart.png)

Figure 5: Human preference for two pairs of models.

### D.3 Data Generation Cost

We compare DIFFIT’s total construction cost with that of several open-source instruction-tuning datasets in Table 6.

A DiffLM with a BART-large decoder is easy to train and implement on consumer-level servers or computers. Compared to self-instruct [6], our method substitutes the LLM-based instruction generation step with DiffLM sampling, reducing the dataset construction cost while enhancing the dataset’s diversity in different aspects. Thanks to the low cost of gpt-3.5-turbo’s API call, we further reduce the cost of output generation.

### E Human Preference Evaluation

We show human preference results in Figure 5. LLaMA 7B instruction-tuned on dataset augmented by DiffTune is favored 42.9% of the times when compared to the counterpart model tuned only on S.A.D, which is only favored 25.4% of the time. Although LLaMA 7B instruction-tuned on S.A.D.+DIFFIT is still far from comparable with ChatGPT, our generated responses are still favored 18.4% of the time, despite the large discrepancy of model size and training cost.

### F Instruction Tuning Experiment Details

Our evaluation closely follows previous general-purpose LLM’s settings [40, 6, 22, 41, 16], where the evaluation covers different aspects of LLM’s general ability as well as instruction-following capability.

**Evaluations on LLM’s general capability.** We compare the same LLM finetuned on different instruction tuning datasets on the following benchmarks: 1) **MMLU** [42] for factual knowledge evaluation, which contains multiple-choice questions from 57 subjects covering different difficulties. 2) **GSM** [43] for mathematical reasoning, which contains grade school-level math problems. 3) **TyDiQA** [44] for multilingual evaluation, which contains machine reading comprehension or question-answering tasks in 11 typologically diverse languages. 4) **Codex-HumanEval** [45] for coding evaluation, which requires the model to generate code given a docstring.

For all benchmarks, we follow the evaluation setting of [16], except that we use Alpaca’s dialogue template instead of Tulu’s.
<table>
<thead>
<tr>
<th>Hyperparameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>#Trainable Params</td>
<td>214M</td>
</tr>
<tr>
<td>Max Seq Length</td>
<td>64</td>
</tr>
<tr>
<td>Diffusion Steps</td>
<td>1000</td>
</tr>
<tr>
<td>Noise Schedule</td>
<td>Linear</td>
</tr>
<tr>
<td>Regression Loss</td>
<td>L1</td>
</tr>
<tr>
<td>DiffLM Transformer Layers</td>
<td>12</td>
</tr>
<tr>
<td>DiffLM Transformer Dim.</td>
<td>768</td>
</tr>
<tr>
<td>Optimizer</td>
<td>AdamW [2019]</td>
</tr>
<tr>
<td>Learning Rate</td>
<td>1e-4</td>
</tr>
<tr>
<td>Batch Size</td>
<td>64</td>
</tr>
<tr>
<td>Warmup Steps</td>
<td>1000</td>
</tr>
<tr>
<td>Learning Rate Schedule</td>
<td>Cosine Decay</td>
</tr>
<tr>
<td>Weight Decay</td>
<td>1e-6</td>
</tr>
<tr>
<td>Dropout</td>
<td>0.0</td>
</tr>
<tr>
<td>Gradient Clipping</td>
<td>1.0</td>
</tr>
<tr>
<td>EMA Decay</td>
<td>0.9999</td>
</tr>
<tr>
<td>Iterations</td>
<td>300K</td>
</tr>
</tbody>
</table>

Table 7: Hyperparameter settings for training and sampling DiffLM.

Evaluations on LLM’s instruction-following capability. We evaluate LLM’s instruction-following capability by comparing different model’s outputs to real-world user inputs. Both automatic and human evaluations are conducted to evaluate the helpfulness and validity of LLM’s response: 1) AlpacaFarm [41] for automatic evaluation on instruction-following capabilities, which uses GPT-4 to compare an LLM’s generation with text-davinci-003’s generation on 805 instructions. 2) VicunaEval [22] for human evaluation on instruction-following capabilities, which contains 80 instructions covering a wide range of scenarios.

Human Evaluation. Human evaluation on VicunaEval covers two aspects: 1) Answer validity, where we ask evaluators to decide whether an LLM’s response is acceptable; and 2) Pairwise comparison, where we ask evaluators to compare two LLM’s response to the same instruction, then score their preference based on a 5-level scale to decide if one response is much better, slightly better or on-par compare to the other.

The instruction for human evaluation comes from Vicuna’s evaluation set [22], which contains 80 instructions covering multiple daily scenarios. We generate responses from LLaMA 7B + S.A.D. (2x), LLaMA 7B + S.A.D. + DIFFIT and ChatGPT for each instruction, and ask human evaluators to 1) Evaluate the validity of the model’s response, and 2) Select the preferred response from two different model’s generations. We conduct two pairs of preference evaluations: LLaMA 7B + S.A.D. + DIFFIT vs. LLaMA 7B + S.A.D. (2x), and LLaMA 7B + S.A.D. + DIFFIT vs. ChatGPT.

The evaluation is conducted with Google Forms. We split each pair of comparisons into two forms, with 40 instructions in each form. In each form, the ordering of the questions and the responses are totally randomized. We recruited 12 experienced annotators who are fluent in English and hold bachelor’s degrees or above. Each form will be assigned to three different evaluators.

We show an example of our evaluation form in Figure 6.

G Experiment Details for DiffLM Training

For all DiffLM experiments, we train the DiffLM on S.A.D.’s training set using the hyperparameter settings shown in Table 7. The generation configuration for the BART decoder is shown in Table 8. Each setting is trained on a single NVIDIA V100 32G.
### Table 8: Generation configuration for DiffLM’s BART decoder.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>max_length</td>
<td>64</td>
</tr>
<tr>
<td>min_length</td>
<td>8</td>
</tr>
<tr>
<td>num_beams</td>
<td>10</td>
</tr>
<tr>
<td>do_sample</td>
<td>False</td>
</tr>
<tr>
<td>no_repeat_ngram_size</td>
<td>2</td>
</tr>
<tr>
<td>repetition_penalty</td>
<td>4.0</td>
</tr>
</tbody>
</table>

We show some of the instances from the DIFFIT Dataset in Table 9. Please note that the table includes examples that collect hallucinating results from ChatGPT, although the answer’s format is aligned with our instruction-tuning objective.
Tell me about the advantages and disadvantages of different programming languages. Add a bullet point summary explaining the pros and cons of each language. Give me a brief explanation of the difference between Swift and Objective-C. Show me how to use C# instead of C++. Tell me why.

Different programming languages have unique advantages and disadvantages. For example, C++ is fast and flexible, but difficult to learn. Java is platform-independent, but slower than some languages. Python has simple syntax, but is slower compared to other languages. Swift is faster than Objective-C, but still somewhat new and has limited library support. C# is a cleaner version of C++ with better memory management, but it is not as fast as C++. To use C# instead of C++, you would need to download and install Microsoft’s Visual Studio, a software development environment. C# is often used for web, Windows, and game development.

Title: “Galactic Rift” In a distant future, humanity has mastered space travel and colonization across multiple galaxies. But as the population soars, a powerful cult arises, vowing to cleanse the universe of all non-believers and establish a new order. Our protagonist, a small group of renegade fighters led by a young woman named Nova, must travel to the cult’s stronghold and stop their plans before it’s too late. Along the way, they will face alien creatures, rogue robots, and the cult’s elite army. As they progress, they discover that the cult is not what they seem - it’s led by ancient beings who seek to resurrect their god and conquer all life in the universe. Nova and her group must decide whether to continue their mission or join forces with the unlikely allies they meet along the way. Through action-packed battles, moral quandaries, and unexpected twists, “Galactic Rift” explores themes of power, faith, and the true meaning of humanity. This edgy sci-fi/fantasy series is perfect for middle schoolers who want an adventure that will leave them on the edge of their seats.

As an AI concept teacher, I explain that AI refers to the simulation of human intelligence in machines to perform tasks that typically require human cognitive abilities. AI can help businesses streamline processes, make data-driven decisions, and improve customer experiences. AI-powered tools can be integrated into a workplace to enhance productivity, automate repetitive tasks, and improve the accuracy of decision-making. As an AI concept user, I have successfully leveraged AI to automate data analysis and improve customer segmentation for a retail client, resulting in a 20% increase in sales.

To prepare a job resume for a Chinese independent IT consultant and an IT contractor who works for the USA as a CPA consultant in China, highlight relevant certifications, degrees, and experience in the technology and finance industry. Emphasize strong communication skills, understanding of Western business practices, and proficiency in Chinese and English. Showcase prior project success, specific technical skills, and collaboration abilities. As an IT consultant or contractor, showcase relevant project management experience and expertise in the latest technologies that support business solutions. Additionally, mention how you have developed and scaled businesses in both the US and China markets. Be brief and clear about your most significant accomplishments, abilities, and contributions to companies you have worked with.

Sure, to learn Kuolang, you need to start by downloading an IDE that supports the language, such as VS Code. Then, you can learn the basics of the language, such as its syntax and data types. There are many resources available, including online courses, documentation, and tutorials. Once you have a basic understanding of the language, practice coding exercises to reinforce your skills. Finally, when you feel confident, try building a project from scratch to solidify your knowledge. Good luck!
Question 2: What if the Beatles had never formed as a band?

Answer A:
If the Beatles had never formed as a band, the music scene in the 1960s and beyond would have been drastically different. It’s likely that other bands and artists would have risen to take their place, but the impact and influence of the Beatles on popular music cannot be overstated. Their innovative sound, thoughtful lyrics, and charismatic personalities helped to shape the direction of music at the time. The Beatles also served as a cultural icon, influencing fashion, art, and societal attitudes. Without the Beatles, the music industry and youth culture may have taken a different direction.

Answer B:
The world would have been different without the Beatles. They were one of the most influential bands in music history and their impact can still be felt today. Their iconic hits like “Yesterday” and “Hey Jude” have become timeless classics, and their influence on pop culture extends far beyond music. Without the Beatles, there may not have been a British Invasion and other similar musical movements that followed. The world may have missed out on some of the most memorable songs and performances of all time. It’s hard to imagine what our world would look like without the Beatles, but we can certainly appreciate them for the incredible legacy they left behind.

Is Answer A a valid response? A valid response should: (1) Answer the question; (2) Have no significant errors; (3) Have no meaningless text (e.g., repetition).

☐ Yes
☐ No

Is Answer B a valid response? A valid response should: (1) Answer the question; (2) Have no significant errors; (3) Have no meaningless text (e.g., repetition).

☐ Yes
☐ No

Out of the two answers, which one do you prefer as a better answer to the question?

1 2 3 4 5

☐ A is clearly better
☐ ☐ ☐ ☐ ☐ B is clearly better

Figure 6: The form used for human evaluation.