JapaGen: Efficient Few/Zero-shot Learning via Japanese Training Dataset Generation with LLM

Takuro Fujii[1,2,*] and **Satoru Katsumata**[3]
[1]Yokohama National University [2]Nomura Research Institute, Ltd. [3]Retrieva, Inc.
tkr.fujii.ynu@gmail.com satoru.katsumata@retrieva.jp

Abstract

Recently some studies have highlighted the potential of Large Language Models (LLMs) as effective generators of supervised training data, offering advantages such as enhanced inference efficiency and reduced costs associated with data collection. However, these studies have predominantly focused on English language tasks. In this paper, we address the fundamental research question: *Can LLMs serve as proficient training data generators for other language tasks?* Specifically, we leverage LLMs to synthesize supervised training data under few-shot and zero-shot learning scenarios across six diverse Japanese downstream tasks. Subsequently, we utilize this synthesized data to train compact models (*e.g.*, BERT). This novel methodology is termed JapaGen. Our experimental findings underscore that JapaGen achieves robust performance in classification tasks that necessitate formal text inputs, demonstrating competitive results compared to conventional LLM prompting strategies.

1 Introduction

Large language models (LLMs) have demonstrated exceptional performance across various natural language processing (NLP) tasks, even with minimal parameter updates (Brown et al., 2020; Kojima et al., 2022). However, the rapid growth in model size, driven by scaling laws (Kaplan et al., 2020), has led to substantial demands for GPU memory and computational resources, making the operation of LLMs prohibitively expensive.

To mitigate these costs, recent studies have investigated the generation of training data using powerful LLMs, followed by training smaller models (e.g., BERT) on the synthesized supervised data (Ye et al., 2022a,b; Yu et al., 2023; Chung et al., 2023a). This approach, termed SuperGen (Supervision Generation Approach) based on prior work (Meng et al., 2022), has demonstrated promising results. The overview of SuperGen is illustrated in Figure 1. SuperGen has been demonstrated to outperform few-shot and zero-shot prompting and few-shot fine-tuning methods in various tasks, effectively reducing both the cost of collecting supervised data and the operational costs of trained models. However, these studies have been limited to English tasks, and thus, the applicability of SuperGen on other language tasks remain uncertain.

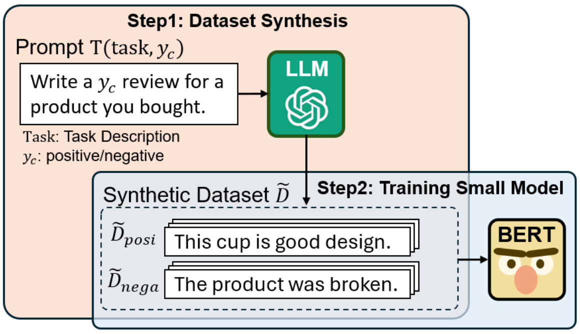

Figure 1: Overview of SuperGen in text sentiment classification as an example.

Given that powerful LLMs like GPT-4 (OpenAI, 2024) are primarily trained on English texts with limited exposure to other languages, it is crucial to investigate the effectiveness of SuperGen in such linguistic contexts and its suitability for different types of languages. In this paper, we implement SuperGen in Japanese as a case study. Japanese is mid-resource language compared to English and has different characteristics, such as the absence of spaces between words. Therefore, we pose the research question: *Do SuperGen methods perform effectively in Japanese?* We term the application of SuperGen to Japanese tasks as JapaGen (§3).

To address the aforementioned interests, we evaluate JapaGen across various Japanese tasks, in-

[*] Work done while internship at Retrieva, Inc. when I was a master student. Now I belong to Nomura Research Institute, Ltd.

cluding text classification, natural language inference, semantic textual similarity, and linguistic acceptability, in both few-shot and zero-shot learning settings. Furthermore, we propose a novel approach termed Knowledge-Assisted Data Generation (KADG)[1], which integrates task-specific knowledge into prompts to align generated texts more closely with gold-standard distributions and enhance text diversity (§3.4).

Our experiments indicate that, in five out of six tasks, zero-shot JAPAGEN outperforms few-shot BERT fine-tuning. Moreover, JAPAGEN demonstrates superior performances in two tasks compared to few-shot PROMPTING. These experimental results suggest that JAPAGEN has the potential to surpass settings with more parameters and more annotated data. Additionally, our analysis shows that KADG enhances the fidelity of generated texts to gold-standard distributions while maintaining label accuracy, although it does not consistently improve overall task performance.

In summary, our contributions are four-fold:

1. We empirically evaluate JAPAGEN, leveraging LLMs as synthetic data generators, across various Japanese NLP tasks.

2. We demonstrate the effectiveness of JAPAGEN, particularly in classification tasks with formal text inputs.

3. We analyze the impact of dataset size on JAPAGEN, observing performance improvements with larger synthetic datasets that eventually reach saturation.

4. We propose and evaluate KADG, demonstrating its potential to refine synthetic data distributions to align with gold standards, thereby enhancing the robustness of JAPAGEN.

2 Related Work

2.1 Efficient Learning Strategies with LLMs

Large Language Models (LLMs) exhibit high performance across various tasks using few-shot or zero-shot learning paradigms. Despite their capabilities, LLMs have numerous parameters, leading to substantial operational costs. To address these challenges, several methods for more efficient utilization of LLMs have been proposed. One such method is PROMPTING, which enables LLMs to perform tasks effectively without requiring parameter updates. This is achieved by injecting prompts based on task descriptions (Brown et al., 2020; Gao et al., 2021; Le Scao and Rush, 2021; Zhang et al., 2022). A prompt consists of input text for the LLM and includes instructions to obtain the desired responses. In few-shot PROMPTING[2], the prompt includes a small number of text-label pairs. Compared to traditional fine-tuning, which necessitates costly updates to the LLM's parameters, PROMPTING improves data efficiency in low-data scenarios. However, Prompting incurs substantial operational costs due to the extensive number of parameters involved.

2.2 Synthesis of Training Data via LLM

To reduce the operational costs of LLMs, researchers have recently explored using LLMs as training data generators, followed by fine-tuning smaller task-specific models (TAMs), such as BERT (Devlin et al., 2019), on the synthetic data. Existing approaches typically employ simple class-conditional prompts and focus on addressing the issues related to the quality of the generated data. Notable early efforts, such as SuperGen (Meng et al., 2022) and ZeroGen (Ye et al., 2022a), have explored the use of LLMs for generating training data for text classification tasks using basic class-conditional prompts. They have also incorporated additional noise-robust learning techniques (Laine and Aila, 2017; Wang et al., 2019) to mitigate the quality issues of the generated data. However, it has been reported that balancing the diversity of synthetic datasets with task performance remains challenging (Chung et al., 2023b).

To date, these approaches have been primarily validated on English-language tasks. This paper investigates the effectiveness of these methods in mid-resource languages with different linguistic characteristics from English.

3 Method: JAPAGEN

In this section, we introduce the motivation for synthetic data generation via LLMs in Japanese tasks, define the problem, and describe the methodology for generating synthetic training data for each task.

[1] We define the setup of KADG as zero-shot* to distinguish it from strict zero-shot methods due to the incorporation of task knowledge.

[2] Few-shot PROMPTING is referred to as In-Context Learning (Brown et al., 2020), however, in this paper, both few-shot and zero-shot PROMPTING are collectively termed as PROMPTING.

The overview of generating training data via LLMs is illustrated in Figure 1.

3.1 Motivation

We define JAPAGEN as the Japanese counterpart to SUPERGEN. The rationale behind selecting Japanese stems from its status as a mid-resource language compared to English, and its different characteristics, such as the absence of spaces between words. Given that powerful LLMs are primarily trained on English texts with limited exposure to other languages including Japanese, it is plausible that they can generate high-quality pseudo training data in English. In this paper, we evaluate JAPAGEN, the Japanese version of SUPERGEN, as a case study focusing on such languages.

3.2 Problem Definition

Given the label space $\mathcal{Y} = \{y_i\}_{i=1}^{n}$, we manually create label-descriptive prompts $\mathrm{T}(\text{task}, y_i)$. For prompt details used in our experiments, please refer to §A.4. We employ LLMs G_θ to generate training data for encoder models E_ϕ (e.g., LSTM (Hochreiter and Schmidhuber, 1997), BERT (Devlin et al., 2019)), which are subsequently fine-tuned as estimators. SUPERGEN comprises the following three stages: (1) Synthesizing supervised training data using LLM. (2) Fine-tuning small models using synthetic data. (3) Testing the trained model on gold data.

3.3 Pseudo Data Generation

In this section, we describe the process of generating pseudo datasets using an LLM for classification and regression tasks. Our approach includes either a single sentence or a sentence pair as input.

Single Sentence Task We employ an LLM to generate pseudo-supervised sentences $\tilde{x}_{c,j}$ corresponding to a label y_c:

$$\tilde{x}_{c,j} \sim \mathrm{Prob}_{\mathrm{LLM}}(\cdot|\mathrm{T}(\text{task}, y_c)), \tag{1}$$

where $\mathrm{T}(\text{task}, y_c)$ represents a prompt including the task description and label y_c. By repeating Equation 1 M times, we obtain the pseudo dataset $\tilde{D}_{y_c} = \{(\tilde{x}_{c,j}, y_c)\}_{j=1}^{M}$. Applying this process for all labels $\{y_c\}_{c=1}^{C}$, we generate the pseudo dataset $\tilde{D} = [\tilde{D}_{y_1}, \tilde{D}_{y_2}, ..., \tilde{D}_{y_C}]$.

Sentence Pair Task Initially, we employ an LLM to generate the first sentence $\tilde{x}_{c,j}^{1}$, analogous to Equation 1 but excluding the label y_c:

$$\tilde{x}_{c,j}^{1} \sim \mathrm{Prob}_{\mathrm{LLM}}(\cdot|\mathrm{T}(\text{task})). \tag{2}$$

In the initial phase of sentence generation, the prompt comprises solely the task description. Subsequently, to generate the second sentence $\tilde{x}_{c,j}^{2}$, the prompt is augmented to include the task description, the first sentence $\tilde{x}_{c,j}^{1}$, and the label y_c:

$$\tilde{x}_{c,j}^{2} \sim \mathrm{Prob}_{\mathrm{LLM}}(\cdot|\mathrm{T}(\text{task}), \mathrm{T}(\text{task}, \tilde{x}_{c,j}^{1}, y_c)). \tag{3}$$

By repeating Equations 2 and 3 M times, we generate the pseudo dataset $\tilde{D}_{y_c} = \{(\tilde{x}_{c,j}^{1}, \tilde{x}_{c,j}^{2}, y_c)\}_{j=1}^{M}$. Applying this process for all labels $\{y_c\}_{c=1}^{C}$, we obtain the pseudo dataset $\tilde{D} = [\tilde{D}_{y_1}, \tilde{D}_{y_2}, ..., \tilde{D}_{y_C}]$.

3.4 Knowledge-Assisted Data Generation

The diversity of synthetic datasets significantly enhances dataset quality, a critical factor in improving task performance (Chung et al., 2023b). Previous studies attempted to diversify text generation by adjusting hyperparameters such as Top-p and temperature. However, this approach may compromise label accuracy. In this paper, we introduce *Knowledge-Assisted Data Generation* (KADG) to enhance dataset diversity while maintaining label correctness.

For each task, we manually create a set of task-specific words S_{task}, and randomly select a word d from this set. We construct a prompt based on the task description, label y_c, and the selected task-specific word d:

$$d \sim S_{\text{task}}, \tag{4}$$

$$\tilde{x}_{c,j} \sim \mathrm{Prob}_{\mathrm{LLM}}(\cdot|\mathrm{T}(\text{task}, y_c, d)). \tag{5}$$

By following a process similar to Section 3.3 across all classes, we generate the synthetic dataset $\tilde{D}$. For the actual prompts used in our experiments, please refer to §A.4.

4 Experiment

In this section, we present an overview of the benchmark datasets, the corresponding evaluation settings, the baseline methods, and the implementation details. Subsequently, we compare our JAPAGEN to baseline methods in both few-shot and zero-shot settings.

4.1 Setup

Benchmarks. To evaluate JAPAGEN across various tasks, we used the following benchmarks from JGLUE (Kurihara et al., 2022): MARC-ja, JSTS, JNLI, and JCoLA. Additionally, to test across diverse domains, we also used two datasets for news topic classification (News) and SNS fact classification (COVID-19). All of these benchmarks are Japanese tasks. JSTS involves sentence similarity estimation, while the others are text classification tasks. We evaluated using Spearman's rank correlation coefficient (Spearman score) for JSTS, Matthews correlation coefficient (MCC; (Matthews, 1975)) for JCoLA, and Accuracy for the remaining tasks. For more detailed information such as dataset statistics and task explanations, please refer to Section A.1.

Baselines. We compared the performances of JapaGen with three baselines: (1) Prompting, a prompt-based learning framework via LLM, as introduced in Section 2.1. (2) Few-Shot Fine-Tuning, where BERT is fine-tuned on five gold samples per class. (3) Fully Supervised, where BERT is fine-tuned on all gold data. We evaluated the performances of JapaGen and Prompting in both few- and zero-shot settings. In the few-shot setting, we used one sample per class and incorporated them into the prompt. To distinguish between the few-shot setting of BERT fine-tuning and the one of JapaGen and Prompting, we refer to the former as "few-shot Ⓑ" and the latter as "few-shot Ⓛ".

Implementation Details. We conducted our experiments using PyTorch (Paszke et al., 2019) and Hugging Face Transformers (Wolf et al., 2020). For synthetic data generation, we utilized the OpenAI model `gpt-3.5-turbo-0613`[3]. The size of the generated data was 25,000 per class. In the few-shot setting Ⓑ, one sample per class was randomly selected. The generation parameters were set to max tokens of 500, top-p of 1.0, temperature of 1.2, and frequency penalty of 0.02, with five pieces of data generated at a time. In JSTS whose labels are continuous values between 0.0 and 5.0, we set six classes {0, 1, 2, 3, 4, 5}. For the fine-tuning of BERT, we used the pretrained BERT[4] and performed our experiments on a single NVIDIA TITAN RTX 24GB GPU. The training parameters[5] were set to batch size of 32, epoch of 4, label smooth temperature of 0.1, optimizer of AdamW with learning rate of 5e-5, β_1 of 0.9, β_2 of 0.999, warmup ratio of 0.1. Additionally, we set max token length of 512, 512, 512, 128, 512, 384 for MARC-ja, JNLI, JSTS, JCoLA, News, and COVID-19 respectively. For each task, we measured performances over five runs with different random seeds. In the few-shot setting Ⓛ, we randomly selected five samples per class.

4.2 Experimental Results

In this section, we compare JapaGen to baselines. Our experimental results are shown in Table 1.

Zero-shot JapaGen vs. Fine-Tuning

Compared to zero-shot JapaGen, BERT fine-tuned on gold data uses the same model size but with a larger amount of annotated data. It is well-known that the zero-shot approach cannot outperform task-specific models trained on human-annotated data. In Table 1, JapaGen adheres to this rule, underperforming compared to fully supervised fine-tuning across all tasks. However, JapaGen outperforms few-shot fine-tuning on five tasks except for COVID-19. Notably in JSTS, JapaGen achieves a Spearman score of 57.67%, exceeding the performance of few-shot Ⓑ fine-tuning. This result suggests that JapaGen can be effective in scenarios where the cost of data collection or annotation is high.

Zero-shot JapaGen vs. Prompting

Compared to zero-shot JapaGen, Prompting employs a significantly larger model size. In Table 1, JapaGen achieves performance improvements of 3.94%, 4.96%, and 17.10% over zero-shot Prompting on JSTS, JNLI, and News, respectively. These tasks typically involve formal text as input. Moreover, JapaGen also surpasses few-shot Ⓛ Prompting on JNLI and News, suggesting that JapaGen has the potential to outperform settings with more parameters and more annotated data. These tasks are commonly classification tasks that involve formal text as input.

KADG and JapaGen

We attempt to enhance the performance of JapaGen by injecting task knowledge into prompts, as prompt engineering has been shown to enhance the capability of LLMs and improve the quality of generated text (Wu and Hu, 2023; Yang et al., 2023; He et al., 2022). In Table 1, KADG outperforms

[3] The generated texts are used solely for study purposes, not for commercial use.

[4] `tohoku-nlp/bert-base-japanese-v3`

[5] We set training parameters based on (Kurihara et al., 2022).

Method	MARC-ja Acc.	JSTS Spearman	JNLI Acc.	JCoLA Mcc.	News Acc.	COVID-19 Acc.	Avg.
FINE-TUNING: *fine-tuning pretrained BERT under gold data.*							
Fully Supervised	95.78±0.1	87.47±0.5	90.19±0.4	40.62±1.2	95.75±0.4	78.49±0.3	82.82
Few-Shot	61.57±8.5	14.80±11.3	37.72±13.4	-0.85±3.5	51.98±5.3	42.24±9.4	37.40
PROMPTING: *prompt-based LLM learning.*							
Zero-Shot	94.82±0.2	68.53±0.6	41.53±1.0	24.76±1.2	40.27±1.3	62.76±0.6	57.66
Few-Shot	97.38±0.2	78.50±2.0	35.86±5.3	26.00±2.9	44.82±2.9	65.44±3.4	61.72
JAPAGEN: *fine-tuning pretrained BERT under pseudo training data generated via LLM.*							
Zero-Shot	77.76±5.4	**72.47**±0.1	**46.49**±1.5	18.17±1.7	**57.37**±2.1	34.36±6.4	54.23
w/ KADG	83.24±6.0	**71.49**±1.2	**46.04**±0.4	16.22±0.5	**59.00**±1.4	26.29±0.8	50.38
Few-Shot	62.97±7.3	**72.56**±0.3	**50.82**±0.8	14.54±1.1	**62.86**±2.8	43.13±1.5	51.15

Table 1: Results on six Japanese tasks. Each value is average with standard deviations over five runs. The tasks that JAPAGEN outperforms zero-shot PROMPTING are in **gray**. Zero-shot JAPAGEN outperforms zero-shot PROMPTING on JSTS, JNLI, ad News. Few-shot (Only one sample per class) JAPAGEN can improve performances on JNLI and News.

zero-shotJAPAGEN only on MARC-ja and News, but does not improve performance on the other four tasks. Specifically, KADG achieves a 5.48% higher score than JAPAGEN on MARC-ja. This suggests that prompt engineering may be particularly effective for specific tasks. In JAPAGEN, the few-shot Ⓛ setting consistently outperforms the zero-shot setting on JSTS, JNLI, News, and COVID-19. Notably, the few-shot setting achieves improvements of 4.33%, 5.49%, and 8.77% over the zero-shot settings on JNLI, News, and COVID-19, respectively. Injecting task knowledge into prompts or using few-shot samples can bring generated texts closer to gold-standard texts, but it may restrict the diversity of the synthetic dataset. A detailed analysis is provided in §4.3.

4.3 Additional Analysis

In this section, we analyze JAPAGEN on distribution, diversity, and label correctness of synthetic and gold datasets. Then, we qualitatively evaluate synthetic data for each task.

Distribution. One of the critical factors influencing task performance is the alignment between the distributions of gold data and synthetic data. To observe this alignment, we compare token appearances within their respective datasets in a simple manner. Figure 2 represents the distribution of token frequencies within the dataset. We also quantitatively assess the alignment using the weighted Jaccard index, based on 1,000 samples per class for distribution analysis. In the top and middle sections of Figure 2, KADG achieves a higher Jaccard index compared to zero-shot JAPAGEN for MARC-ja, JSTS, JNLI, and News. Conversely, in the top and bottom sections of Figure 2, few-shot JAPAGEN outperforms zero-shot JAPAGEN regarding the Jaccard index for JSTS, JNLI, and News. Qualitatively, we observe a decrease in the number of words appearing only in the synthetic dataset, the blue-only part in Figure 2, with KADG and the few-shot setting. These results suggest that designing effective prompts and incorporating a few real samples can help bring the synthetic data distribution closer to that of the gold standard.

Diversity & Label Correctness. Synthetic datasets often exhibit limited diversity because they are generated using the same prompt input into the LLM. To assess dataset diversity, we adopt the methodology of a previous study (Holtzman et al., 2020) and use the Self-BLEU metric (Zhu et al., 2018) to compare the diversity of synthetic and gold datasets. A lower Self-BLEU score indicates higher dataset diversity. Previous studies have highlighted a trade-off between dataset diversity and label correctness (Chung et al., 2023b; Ye et al., 2022a). Consequently, we also evaluate label correctness in the synthetic dataset. To do so, we first train BERT on the gold training dataset and then measure accuracy[6] on the synthetic dataset. Table 2 presents the diversity and label correctness analysis for each task.

[6] In JSTS, Mean Squared Error (MSE) is used for measurement.

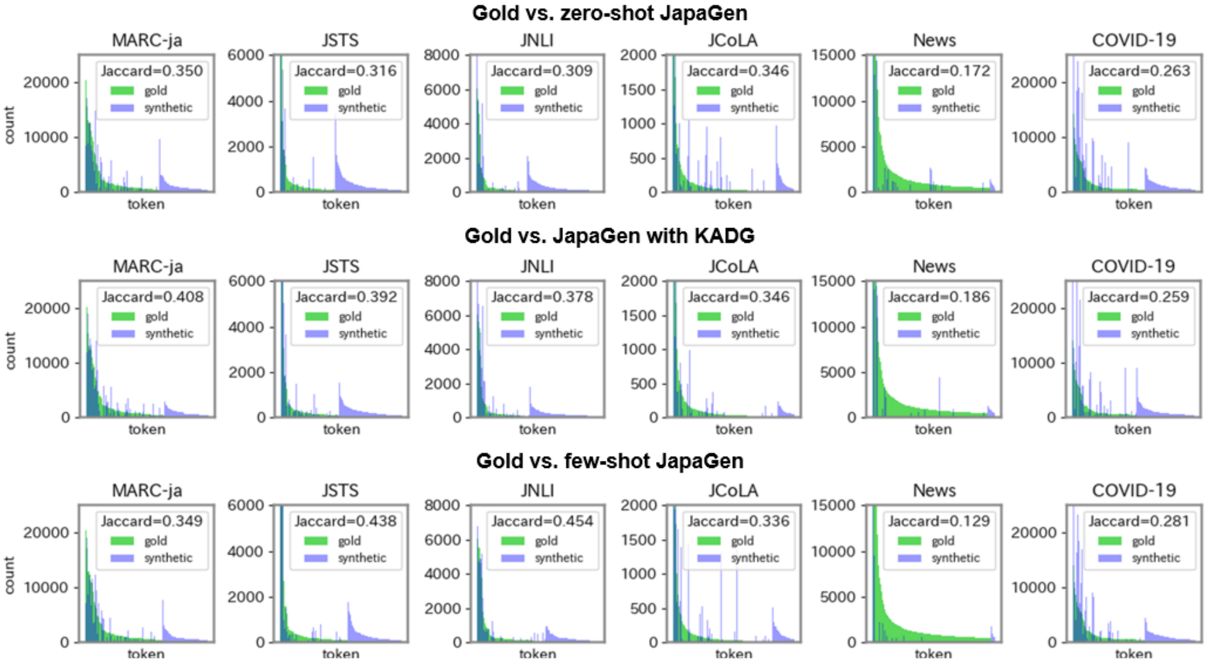

Figure 2: Distribution of the number of appeared tokens between gold and synthetic dataset. Top: zero-shot JapaGen, Middle: JapaGen with KADG, and Bottom: few-shot JapaGen. Compared to zero-shot JapaGen, KADG can improve alignment between gold and synthetic dataset on MARC-ja, JSTS, JNLI, and News. Few-shot JapaGen can also improve alignment on JSTS, JNLI, and COVID-19.

Dataset	MAR.	JSTS*	JNLI	JCoLA
		Diversity (%)		
Gold	40.53	72.93	72.94	56.66
Zero-shot	91.67	74.89	69.97	65.80
w/ KADG	84.97	76.12	73.13	78.91
Few-shot	90.25	81.80	78.28	67.15
		Label Correctness (%)		
Gold	99.06	0.137	98.01	96.28
Zero-shot	99.97	1.540	35.11	66.34
w/ KADG	99.96	1.540	39.37	63.94
Few-shot	99.90	1.094	50.16	63.33

Table 2: Diversity and label correctness of synthetic dataset. We measure the diversity by Self-BLEU. *In JSTS, label correctness is measured by MSE.

As shown in the upper part of Table 2, the Self-BLEU score of the synthetic dataset of zero-shot JapaGen is approximately twice as high, indicating less diversity compared to the gold dataset in MARC-ja. However, zero-shot JapaGen can synthesize datasets with a diversity similar to the gold dataset in JSTS, JNLI, and JCoLA. In contrast, in the lower part of Table 2, the label correctness in JSTS, JNLI, and JCoLA is not as high as in the gold dataset. Despite reports suggesting that decreasing the Self-BLEU score reduces label accuracy and degrades downstream task performance (Ye et al., 2022a), in MARC-ja, KADG improves the Self-BLEU score without compromising label correctness and enhances downstream performance. The few-shot setting yielded results similar to zero-shot JapaGen in diversity, but improvements in label correctness were observed in the two tasks, JSTS and JNLI.

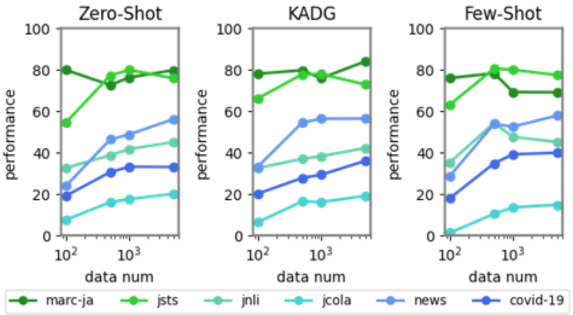

Figure 3: Performance transition with synthetic dataset size on zero-shot, KADG, and few-shot settings.

Data Scaling. We analyze the performance scaling with respect to data size. Figure 3 demonstrates that for most tasks, performance improves as the data size increases. However, performance tends to plateau, as the results with 5,000 samples are similar to those with 50,000 samples.

Task	Synthesized Text	Label
MARC-ja	この商品は思っていた以上に素晴らしかったです！購入して本当に良かったです。... (This product was even more nice than I expected! I’m really glad I bought it. ...)	Positive
	商品は非常に不満でした。品質が悪い上に、配送にも遅延がありました。使ってみると... (I was extremely dissatisfied with the product. In addition to poor quality, there were delays in delivery. ...)	Negative
JSTS	子供たちが講演で楽しそうに遊んでいます。 (The children are having fun playing in the park.) 講演で遊ぶ子供たちが笑顔で何かを楽しんでいます。 (The children playing in the park are smiling and enjoying something.)	similarity = 1.0
JNLI	幸せそうなカップルが手をつないで海辺を歩いている。 (A happy couple is walking hand in hand along the seaside.) 青い空と波が背景に広がり、夕日の光が二人を照らしている。 (With the blue sky and waves in the background, the light of the setting sun shines on the couple.)	Entailment
	木々が繁茂する森の中で、明るい光が差し込む風景。 (In the forest where trees grow thickly, bright light streams through the landscape.) 濃い霧がかかり、視界がほとんどない中に立つ孤独な木。 (A solitary tree stands amidst a dense fog, with almost no visibility.)	Contradiction
	美しい夕焼け空の中、風景画の中に描かれた山々の輪郭が静かに浮かび上がっている。 (In the beautiful sunset sky, the outlines of mountains depicted in the landscape painting quietly emerge.) 夕暮れ時に描かれた風景で、美しく彩られた空の中には山々の輪郭が描かれています。 (In the landscape painted at dusk, the outlines of mountains are depicted against a beautifully colored sky.)	Neutral
JCoLA	私は友達と昨日食べった寿司にします。 (I will have the sushi I atte with my friends yesterday.)	Unacceptable
	昨日の夜、友達とおいしいラーメンを食べました。 (Last night, I ate delicious ramen with my friends.)	Acceptable
COVID-19	COVID-19の最新情報です。感染拡大を防ぐためには、手洗いやマスクの着用、人との距離... (Here is the latest information on COVID-19. To prevent the spread of infection, it is important,...)	General Fact
	今日は友人がCOVID-19に感染していました。心配ですが、早く回復することを... (Today, my friend tested positive for COVID-19. I’m worried, but I hope they recover quickly...)	Personal Fact
	新型コロナウイルスの感染が拡大する中、マスクの着用や手洗いの重要性を再認識し... (Amid the spread of the novel coronavirus, I have come to realize once again the importance...)	Opinion
	今日はおいしいお寿司を食べました！旬のネタが特に美味しかったです！ (Today, I had some delicious sushi! The seasonal toppings were especially tasty!)	Impression
News	日本の低価格航空会社PeachAviationは、ユーザーにより快適なフライト体験を提供するための新しい取り組みを発表しました。 (Japan’s low-cost airline Peach Aviation has announced a new initiative to provide users with a more comfortable flight experience.)	Peachy
	日本の航空会社、エスマックスが業績好調であることが報じられました。新たな路線の開設や購入した新型機の稼働により、利益が大幅に上昇しています。 (It has been reported that Japan’s airline, Smax, is experiencing strong performance. The opening of new routes and the operation of newly purchased aircraft have significantly increased their profits.)	S-MAX

Table 3: Synthesized data sample by zero-shot JapaGen for each task.

4.4 Qualitative Evaluations

We observe that JapaGen was generally able to synthesize texts in accordance with the tasks. Below, we describe examples where JapaGen did not perform well for each task.

MARC-ja. JapaGen tends to generate similar texts such as "この商品は良い/悪いです。(This commodity is good/bad.)". Table 2 also indicates a high Self-BLEU score for MARC-ja, implying significant similarity among the synthesized texts. As indicated by the high score of label correctness in Table 3, we observe no discrepancy between the synthesized text and the corresponding label.

JSTS. While labels are continuous values, employing discrete values as labels in the prompt limits the capability of JapaGen to capture detailed similarity between two sentences. For instance, the similarity between the two sentences presented in Table 3 is 1.0. However, from the perspective of native Japanese speakers, this similarity should be rated above 3.0. The label correctness score (MSE) of synthesized texts by JapaGen is also too high, which suggests that several labels are not correct, compared to that of gold data.

JNLI. JapaGen exhibits difficulty distinguishing between "Entailment" and "Neutral". Specifically, text pairs for "Neutral" are frequently misclassified as "Entailment". The label correctness score (Accuracy) of synthesized texts by JapaGen is also too low compared to that of the gold data.

JCoLA. JCoLA is a binary classification task to predict whether a Japanese text is syntactically acceptable or unacceptable. Our observation indicate that the LLM struggles with generating unacceptable sentences. Specifically, the expression "食べった" in Table 3 is not a syntactic error but a typo. This is because LLMs are trained to generate syntactically correct sentences, leading to difficulties in generating grammatically incorrect ones.

COVID-19. Synthesized texts correspond to each label; however, JapaGen frequently generates similar texts (*e.g.*,"手洗い" (washing hands), "マスク" (wearing a mask)) within a label. The Self-BLEU score of synthetic texts in COVID-19 is much higher, indicating lower diversity compared to gold data presented in Table 5.

News. This is a news topic classification task where topic names as labels include entity-like unique expressions. Synthetic texts frequently fail to align with these labels, particularly when the labels involve proper nouns or lacks common sense. For instance, in Table 3, "Peachy" is a category indicating news targeting women; however, it generates content about the real airline "Peach (Peach Aviation)". Similarly, "S-MAX" is a category for software-related news; however, it frequently produces content about fictional people or companies named 'S-MAX' are often generated.

Throughout all six tasks, while the text synthesized by JapaGen has challenges in terms of diversity and label consistency, it was generally able to produce text that aligned with the tasks.

4.5 Overall Results

In this section, we summarize §4.2, §4.3, and §4.4 related to the experimental results and analysis. The results of zero-shot JapaGen, comparing to few-shot fine-tuning and prompting, showed that it is particularly effective for classification tasks with formal text input. This suggests JapaGen has the potential to surpass scenarios with more parameters and more annotations. Additionally, the results from KADG and few-shot JapaGen indicated that incorporating task knowledge and examples into the prompts can further enhance its capabilities. On the other hand, challenges include low label correctness and the difficulty in synthesizing datasets with continuous value labels such as JSTS and with the desired grammatical errors in JCoLA.

5 Conclusion

To investigate the effectiveness of SuperGen in a mid-resource language with characteristics different from English, we evaluated SuperGen specifically for Japanese tasks, termed JapaGen. Our experimental results demonstrate that JapaGen is particularly effective for classification tasks where the input consists of formal text compared to few-shot Prompting.

Future Work

- We will examine the efficacy of prompts in synthesizing high-quality texts for specific tasks.
- As the development of open LLMs is also progressing rapidly, we would like to evaluate JapaGen using such LLMs.

Limitation

- Our trained models are unavailable for commercial use because we used OpenAI LLM for data generation.
- Although we used GPT-3.5 as a pseudo training data generator, using more advanced LLM (*e.g.*, GPT-4) might yield different results.
- To examine the impact of SuperGen on languages with distinct characteristics from English and classified as mid-resource, we selected Japanese as a case study. Future research will address additional languages.

Ethics Statement

While PLMs have demonstrated remarkable capabilities in text generation and comprehension, they also pose potential risks or harms (Bender and Koller, 2020; Bender et al., 2021), such as generating misinformation (Pagnoni et al., 2021) or amplifying harmful biases (Prabhumoye et al., 2018). Our work specifically focuses on leveraging existing PLMs to generate training data for NLU tasks, rather than on developing new PLMs or generation methods. In this study, we comply with the OpenAI's terms of use by not disclosing synthetic data and by refraining from using it for purposes other than study. Furthermore, this study did not involve any sensitive data but only used publicly available data, including MARC-ja, JSTS, JNLI, JCoLA, News, and COVID-19.

References

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? In Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, page 610–623.

Emily M. Bender and Alexander Koller. 2020. Climbing towards NLU: On meaning, form, and understanding in the age of data. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 5185–5198, Online. Association for Computational Linguistics.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In Advances in Neural Information Processing Systems, volume 33, pages 1877–1901. Curran Associates, Inc.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollar, and C. Lawrence Zitnick. 2015. Microsoft coco captions: Data collection and evaluation server.

John Chung, Ece Kamar, and Saleema Amershi. 2023a. Increasing diversity while maintaining accuracy: Text data generation with large language models and human interventions. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 575–593, Toronto, Canada. Association for Computational Linguistics.

John Chung, Ece Kamar, and Saleema Amershi. 2023b. Increasing diversity while maintaining accuracy: Text data generation with large language models and human interventions. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 575–593, Toronto, Canada. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 3816–3830, Online. Association for Computational Linguistics.

Yun He, Steven Zheng, Yi Tay, Jai Gupta, Yu Du, Vamsi Aribandi, Zhe Zhao, Yaguang Li, Zhao Chen, Donald Metzler, Heng-Tze Cheng, and Ed H. Chi. 2022. HyperPrompt: Prompt-based task-conditioning of transformers. In Proceedings of the 39th International Conference on Machine Learning, volume 162 of Proceedings of Machine Learning Research, pages 8678–8690. PMLR.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. Neural Comput., 9(8):1735–1780.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In International Conference on Learning Representations.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. CoRR, abs/2001.08361.

Phillip Keung, Yichao Lu, György Szarvas, and Noah A. Smith. 2020. The multilingual Amazon reviews corpus. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 4563–4568, Online. Association for Computational Linguistics.

Takeshi Kojima, Shixiang (Shane) Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In Advances in Neural Information Processing Systems, volume 35, pages 22199–22213. Curran Associates, Inc.

Kentaro Kurihara, Daisuke Kawahara, and Tomohide Shibata. 2022. JGLUE: Japanese general language understanding evaluation. In Proceedings of the Thirteenth Language Resources and Evaluation Conference, pages 2957–2966, Marseille, France. European Language Resources Association.

Samuli Laine and Timo Aila. 2017. Temporal ensembling for semi-supervised learning. In International Conference on Learning Representations.

Teven Le Scao and Alexander Rush. 2021. How many data points is a prompt worth? In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 2627–2636, Online. Association for Computational Linguistics.

B.W. Matthews. 1975. Comparison of the predicted and observed secondary structure of t4 phage lysozyme. Biochimica et Biophysica Acta (BBA) - Protein Structure, 405(2):442–451.

Yu Meng, Jiaxin Huang, Yu Zhang, and Jiawei Han. 2022. Generating training data with language models: Towards zero-shot language understanding. In Advances in Neural Information Processing Systems, volume 35, pages 462–477. Curran Associates, Inc.

Takashi Miyazaki and Nobuyuki Shimizu. 2016. Cross-lingual image caption generation. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1780–1790, Berlin, Germany. Association for Computational Linguistics.

OpenAI. 2024. Gpt-4 technical report.

Artidoro Pagnoni, Vidhisha Balachandran, and Yulia Tsvetkov. 2021. Understanding factuality in abstractive summarization with FRANK: A benchmark for factuality metrics. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 4812–4829, Online. Association for Computational Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In Advances in Neural Information Processing Systems, volume 32. Curran Associates, Inc.

Shrimai Prabhumoye, Yulia Tsvetkov, Ruslan Salakhutdinov, and Alan W Black. 2018. Style transfer through back-translation. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 866–876, Melbourne, Australia. Association for Computational Linguistics.

Taiga Someya, Yushi Sugimoto, and Yohei Oseki. 2024. JCoLA: Japanese corpus of linguistic acceptability. In Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), pages 9477–9488.

Yisen Wang, Xingjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey. 2019. Symmetric cross entropy for robust learning with noisy labels. In IEEE International Conference on Computer Vision.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pages 38–45, Online. Association for Computational Linguistics.

Yangjian Wu and Gang Hu. 2023. Exploring prompt engineering with GPT language models for document-level machine translation: Insights and findings. In Proceedings of the Eighth Conference on Machine Translation, pages 166–169, Singapore. Association for Computational Linguistics.

Li Yang, Qifan Wang, Jingang Wang, Xiaojun Quan, Fuli Feng, Yu Chen, Madian Khabsa, Sinong Wang, Zenglin Xu, and Dongfang Liu. 2023. MixPAVE: Mix-prompt tuning for few-shot product attribute value extraction. In Findings of the Association for Computational Linguistics: ACL 2023, pages 9978–9991, Toronto, Canada. Association for Computational Linguistics.

Jiacheng Ye, Jiahui Gao, Qintong Li, Hang Xu, Jiangtao Feng, Zhiyong Wu, Tao Yu, and Lingpeng Kong. 2022a. ZeroGen: Efficient zero-shot learning via dataset generation. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 11653–11669, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jiacheng Ye, Jiahui Gao, Zhiyong Wu, Jiangtao Feng, Tao Yu, and Lingpeng Kong. 2022b. ProGen: Progressive zero-shot dataset generation via in-context feedback. In Findings of the Association for Computational Linguistics: EMNLP 2022, pages 3671–3683, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yue Yu, Yuchen Zhuang, Rongzhi Zhang, Yu Meng, Jiaming Shen, and Chao Zhang. 2023. ReGen: Zero-shot text classification via training data generation with progressive dense retrieval. In Findings of the Association for Computational Linguistics: ACL 2023, pages 11782–11805, Toronto, Canada. Association for Computational Linguistics.

Ningyu Zhang, Luoqiu Li, Xiang Chen, Shumin Deng, Zhen Bi, Chuanqi Tan, Fei Huang, and Huajun Chen. 2022. Differentiable prompt makes pre-trained language models better few-shot learners. In International Conference on Learning Representations.

Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. 2018. Texygen: A benchmarking platform for text generation models. In The 41st international ACM SIGIR conference on research & development in information retrieval.

A Appendix

A.1 Dataset and Task

We describe the six tasks used in our experiment. The dataset statistics are presented in Table 4.

MARC-ja A binary classification task to predict the sentiment of product reviews as positive or negative. The dataset used for this task is derived from the Japanese subset of the Multilingual Amazon Reviews Corpus (MARC) (Keung et al., 2020).

JSTS A regression task to predict the semantic similarity score between two sentences. The score ranges from 0 (least similar) to 5 (most similar). The data for this task are sourced from the Japanese version of the MS COCO Caption Dataset (Chen et al., 2015) and the YJ Captions Dataset (Miyazaki and Shimizu, 2016).

JNLI A three-way classification task to predict the relation between two sentences. The possible relations are {contradiction, neutral, entailment} reflecting the categories utilized in the Stanford Natural Language Inference (SNLI) dataset (Bowman et al., 2015). The data source for this task is the same as that used for JSTS.

JCoLA A binary classification task to predict whether a Japanese text is syntactically acceptable or unacceptable. For further details, please refer to (Someya et al., 2024).

News A nine-way classification task to predict the news topic of a given news text. The news texts are sourced from Livedoor News. The possible topics are {Trend Topic News, Sports Watch, IT Life hack, Consumer Electronics, MOVIE, DOKUJOTSUSHIN, S-MAX, HOMME, Peachy}.

COVID-19 A four-way classification task to predict the factuality of tweets about COVID-19. The categories of factual information include "general fact," "personal fact," "opinion," and "impressions." The data for this task are sourced from `https://www.db.info.gifu-u.ac.jp/covid-19-twitter-dataset/`.

A.2 Metrics

Spearman's Correlation Score This metric means the consistency between two sets of rankings by calculating the correlation between their ranks. A score close to 1 indicates strong agreement, meaning the model's ranked outputs closely match the true ranked labels.

Dataset		**Number of Samples** Train	 Dev.	 Test
JGLUE	MARC-ja	150,022	37,506	5,654
	JSTS	9,960	2,491	1,457
	JNLI	16,058	4,015	2,434
	JCoLA	4,000	1,000	865
News		4,375	625	1,475
COVID-19		4,375	625	7,547

Table 4: Dataset statistics.

Dataset	**News**	**COVID-19**
DIVERSITY (%)		
Gold	62.97	43.14
Zero-shot	79.90	84.31
w/ KADG	82.93	81.91
Few-shot	79.25	83.40
LABEL CORRECTNESS (%)		
Gold	98.89	90.87
Zero-shot	49.84	60.80
w/ KADG	43.61	58.86
Few-shot	57.33	64.43

Table 5: Diversity and label correctness of synthetic dataset in News and COVID-19.

Matthews Correlation Coefficient (MCC) MCC measures the quality of binary classifications by considering true positives, false positives, true negatives, and false negatives in a balanced way. Its value ranges from -1 to 1, where 1 indicates perfect prediction, and -1 a complete inverse relationship.

Self-BLEU This metric calculates BLEU scores for generated text samples against other samples within the same set to measure diversity. Lower Self-BLEU indicates more diverse outputs.

A.3 Additional Results

The diversity (Self-BLEU) and label correctness of News and COVID-19 are shown in Table 5. While the diversity of News and COVID-19 in few-shot is lower than that in zero-shot, few-shot JapaGen can improve the label correctness of News and COVID-19.

A.4 Prompt for Each Task

For prompt details used in our experiments, please refer to `https://github.com/retrieva/JapaGen` due to the page limitation.