Bridging the Data Gap: Using LLMs to Augment Datasets for Text Classification

Seyed Parsa Neshaei
EPFL, Lausanne, Switzerland
seyed.neshaei@epfl.ch

Richard Lee Davis
KTH Royal Institute of Technology, Stockholm, Sweden
rldavis@kth.se

Paola Mejia-Domenzain
EPFL, Lausanne, Switzerland
paola.mejia@epfl.ch

Tanya Nazaretsky
EPFL, Lausanne, Switzerland
tanya.nazaretsky@epfl.ch

Tanja Käser
EPFL, Lausanne, Switzerland
tanja.kaeser@epfl.ch

ABSTRACT

Deep learning models for text classification have been increasingly used in intelligent tutoring systems and educational writing assistants. However, the scarcity of data in many educational settings, as well as certain imbalances in counts among the annotated labels of educational datasets, limits the generalizability and expressiveness of classification models. Recent research positions LLMs as promising solutions to mitigate the data scarcity issues in education. In this paper, we provide a systematic literature review of recent approaches based on LLMs for generating textual data and augmenting training datasets in the broad areas of natural language processing and educational technology research. We analyze how prior works have approached data augmentation and generation across multiple steps of the model training process, and present a taxonomy consisting of a five-stage pipeline. Each stage covers a set of possible options representing decisions in the data augmentation process. We then apply a subset of the identified methods to three educational datasets across different domains and source languages to measure the effectiveness of the suggested augmentation approaches in educational contexts, finding improvements in overall balanced accuracy across all three datasets. Based on our findings, we propose our pipeline as a conceptual framework for future researchers aiming to augment educational datasets for improving classification accuracy[1].

Keywords

Data Augmentation, Large Language Models, Fine-tuning, Natural Language Processing, Text Classification

[1]The open-source code of our experiments, as well as the prompts used for the LLM and the detailed results of our experiments, can all be found on:
https://github.com/epfl-ml4ed/data-aug-education

Seyed Parsa Neshaei, Richard Lee Davis, Paola Mejia-Domenzain, Tanya Nazaretsky, and Tanja Kaser. Bridging the Data Gap: Using LLMs to Augment Datasets for Text Classification. In Caitlin Mills, Giora Alexandron, Davide Taibi, Giosuè Lo Bosco, and Luc Paquette (eds.) Proceedings of the 18th International Conference on Educational Data Mining, Palermo, Italy, July, 2025, pp. 119–132. International Educational Data Mining Society (2025).

https://doi.org/10.5281/zenodo.15870195

1. INTRODUCTION

During recent years, there has been a surge in the use of artificial intelligence (AI)-based models, such as deep learning models, within educational technology, particularly in intelligent tutoring systems (ITS) and interactive writing assistants designed to support students in their learning processes [26, 18, 42, 27].

In many of these systems, *text classification*, defined as the ability to categorize the input text, plays a central role. Prior works have explored embedding text classification models in educational tools to support students by determining the proficiency levels of their essays [25], mining components of argumentative and legal texts [63, 60], and also detecting non-content-driven traits in texts such as sentiment [39]. A key enabler of such capabilities lies in deep learning models trained or fine-tuned for text classification, where transformer-based architectures (including BERT [12], DistilBERT [51], or RoBERTa [35]) have achieved unprecedented results on a set of benchmarks regarding model accuracy while staying efficient enough to run on modern hardware.

However, a major limitation of using text classification models in educational settings arises when considering the *datasets* on which the models should be trained. When collecting and labeling data in real-world educational contexts, privacy concerns regarding data collection [48] or domain-specific challenges such as the difficulty of the annotation task and the potential need for annotation experts can severely limit the amount of available data. Moreover, researchers developing educational technology tools also have to address the issue of *class imbalance*, in which certain annotation classes are found more abundantly than the rest in the input dataset [56], shown to be the case across numerous educational datasets [41, 63, 59, 11]. The class imbalance issue can prevent the classification models from reasonably capturing minority classes and might lead to suboptimal generalization

and reduced accuracy in real-world educational settings.

The literature on natural language processing (NLP) has traditionally sought to address the issues mentioned above through classical data augmentation methods (e.g., synonym replacement and back-translation) to introduce linguistic variability into existing datasets [10, 34]. While researchers have shown improvement in model accuracies after applying such traditional methods, their relatively simplistic manipulations of the input data can lead to repetitive or almost-repetitive data samples, limiting the model from being able to learn from new data properly.

Recent large language models (LLMs) such as GPT-4o [19] or Llama [15] have shown strong contextual understanding and generation capabilities. Thus, they can generate semantically coherent data entries in a data augmentation pipeline. LLMs also enable zero-, one-, and few-shot learning approaches, where the model can be applied to a new task with a minimal amount of new training data [64], which is particularly interesting in an educational context with limited data. As a result, there is a growing interest in exploring *LLM-driven* data augmentation methods to increase the quantity of the training datasets and to address the class imbalance issues. Furthermore, there has been an increase in research on *distilling* the knowledge of LLMs into smaller, more efficient, and task-specific classifiers that can be deployed at scale in ITS [32, 33, 13].

Motivated by these recent advancements, in this paper, we conduct a comprehensive and systematic review of the recent research on LLM-driven data augmentation across the domains of NLP and educational technologies. More specifically, we aim to answer the following two research questions:

- **RQ1:** What are the recent LLM-driven data augmentation approaches in the NLP literature, and how can they be categorized?
- **RQ2:** How do recent LLM-based approaches of data augmentation and generation apply across different educational datasets in terms of improving classification accuracy?

To answer these research questions, we first perform a systematic literature review and propose a taxonomy of data augmentation methods. Our taxonomy comes in the form of a pipeline capturing the main components of the data augmentation process discussed in prior literature. We present our pipeline as a *conceptual framework* that can be used and adapted by researchers and practitioners, including those working on models trained on educational datasets.

In the second step, we select a subset of methods from our pipeline and apply them to three distinct educational datasets covering different task definitions (reflective writing and persuasive writing) and languages (English and German), enabling us to assess the effectiveness of state-of-the-art data augmentation methods on educational data.

Our results provide insights into the potential trade-offs and best practices of LLM-driven data augmentation for text classification in educational settings. We offer our pipeline as a recommendation for future researchers looking to utilize the knowledge encoded in domain-independent LLMs to improve the classification accuracy of task-specific models, mitigate the issues of data scarcity and imbalance, and enhance the performance of AI-enabled educational systems.

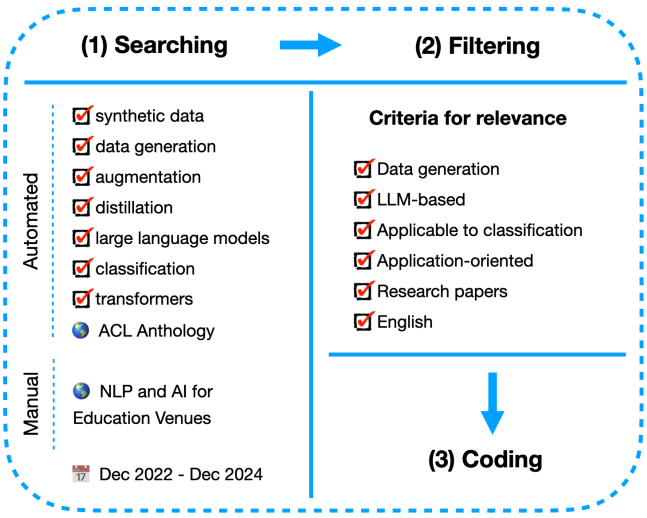

Figure 1: Overview of our literature review process, including automated and manual searching (Step 1), filtering based on relevance criteria (Step 2), and coding (Step 3).

2. DATA AUGMENTATION - A TAXONOMY

We conducted a systematic literature review following the recommendations of [22] as well as the prior literature reviews in the domain of education [54, 40, 24] as a basis for creating a taxonomy of data augmentation methods.

2.1 Literature Review

The structure of our literature review is illustrated in Fig. 1. It consisted of three main steps: searching, filtering, and coding.

2.1.1 Searching

For the automated search, we searched among the content of the relevant scholarly articles in the ACL Anthology database. We used the Google Advanced Search functionality to search among the pages belonging to the "https://aclanthology.org" URL domain. Particularly, we used the following queries: ("synthetic data" OR "data generation" OR "augmentation" OR "distillation") AND "large language models" AND "classification" AND "transformers". We only included papers from December 2022 to December 2024. The search was conducted in December 2024 and January 2025. Our automated search retrieved 1860 entries, from which we discarded redundant pages or pages not referring to a published paper. In addition to the automated search, we performed a manual search in the journals and conferences of NLP and AI in education research communities (e.g., EDM and AIED) to retrieve papers focusing on educational datasets, as well as to find potentially relevant posters or workshop papers.

2.1.2 Filtering

We evaluated each retrieved paper to determine its relevance to the current work. Our criteria for relevance included:

- *Data Generation:* The paper either focuses on data generation, or uses approaches (e.g., data augmentation or distilling LLM knowledge into smaller models), which generate data as part of their process.
- *LLM-based:* The paper uses LLMs for the augmentation or generation task.
- *Downstream Task:* The paper either addresses the task of text classification, or the presented approaches are suitable for text classification.
- *Application:* The paper follows the direction of experimentally evaluating the introduced approach, rather than merely introducing new theory underpinnings or architectures.
- *Research Papers:* We only included research papers in our criteria for relevance; i.e., we excluded non-research papers, literature reviews, and research reports.
- *Language:* We only included papers written in English.

After the filtering process, we included 78 papers in our literature review.

2.1.3 Coding

We first coded each paper based on the NLP task (Classification or Generation) that it addressed. Furthermore, to answer our first research question, i.e., to categorize the recent LLM-based approaches for data augmentation, we developed a data augmentation pipeline consisting of five main stages and coded each paper according to the stages covered, as well as the methods applied within each stage. We created the pipeline using a bottom-up approach from the data by reading the filtered papers from our literature review process. Then, the coding of the papers across each stage, as well as the adaptation of the stages, was conducted by first choosing 5 papers at random and having two coders categorize them across all of the defined stages. We achieved an inter-rater agreement of $\kappa = 0.67$, indicating a *substantial* agreement [37]. One of the two coders then annotated the remaining papers. For each stage, if relevant, we allowed the value of *Other* as well, to be able to include papers not discussing a certain part of the pipeline, or using unique approaches for a certain stage. We also allowed selecting multiple values for each stage, if meaningful.

2.2 Resulting Categorization (RQ1)

Figure 2 illustrates the resulting taxonomy of LLM-based approaches to data augmentation (RQ1), including the number of papers within each category. A complete list of all papers with their respective coding can be found on our GitHub.

2.2.1 Purpose

Stage 0 in Fig. 2 refers to the addressed downstream task. We found that 53% of the analyzed papers (e.g., [8, 74, 46]) focus on *Text Classification*, with the aim to leverage data augmentation to improve the performance of a text classification model. For example, [31] explored synthetic data generation as a means to improve classification accuracy. Another work [7] demonstrated that their method outperforms few-shot LLM-based text classification.

On the other hand, 46% of the papers focus on *Generation*, aiming to improve models generating text (e.g., information extraction, question-answering, solving math word problems, commonsense reasoning, etc.) [32, 1, 71]. For example, [2] have explored fallacy recognition as a question-answering-style task and [20] have focused on the task of event extraction.

2.2.2 Pipeline Stages

Our review of the papers led to the extraction of five common stages in our data augmentation pipeline: 1) Initial Augmentation and Generation, 2) Example Selection, 3) Augmentation based on Examples, 4) Adaptation, and 5) Iterative Loop.

1) Initial Augmentation and Generation. 76% of all papers perform an initial set of data generation and augmentation, before training the model with any data. Within this stage, four main methods are commonly used:

- *Zero- or one-shot:* 29 papers apply zero- or one-shot prompting to generate data, merely describing the desired type of output, or directly applying a transformation on one input data point without in-context learning from several examples (e.g., paraphrasing a sentence)[2]. For example, [6] used a zero-shot prompt for generating a dataset for the task of sentiment classification, [78] explored zero-shot learning for mitigating distribution bias, and [29] augmented data for stance detection using a zero-shot-based approach.
- *Few-shot:* 19 papers in this stage apply few-shot prompting to generate data, providing a set of input data points to the LLM to generate new data points by looking at the overall set of provided shots at each step. For example, [52] used few-shot prompting on five state-of-the-art LLMs to generate high-quality educational questions belonging to different cognitive levels, while [49] used few-shot prompts for generating math explanations.
- *Chain-of-Thought (CoT):* Following the initial idea provided by [65], 6 papers in the first stage include a prompt asking the LLM to think first and then respond, or to identify a list of steps that the LLM should follow before providing the answer in the output. [5], for example, used the CoT outputs of a model to distill knowledge into a conversational agent. [16] introduced a unified framework based on CoT distillation to mitigate the challenge of treating tokens of different significance in the same way.
- *Fine-tuning a model:* Finally, 14 papers in this stage fine-tune an LLM on the training set, and use approaches such as controlled text generation to generate sentences from the desired classes on the fly from the fine-tuned LLM. [4], for instance, fine-tuned a pre-trained generation model on

[2]While certain approaches, such as paraphrasing a sentence, are considered *one-shot*, we grouped them together with zero-shot as opposed to few-shot, because no in-context learning or data mixture from several examples was conducted.

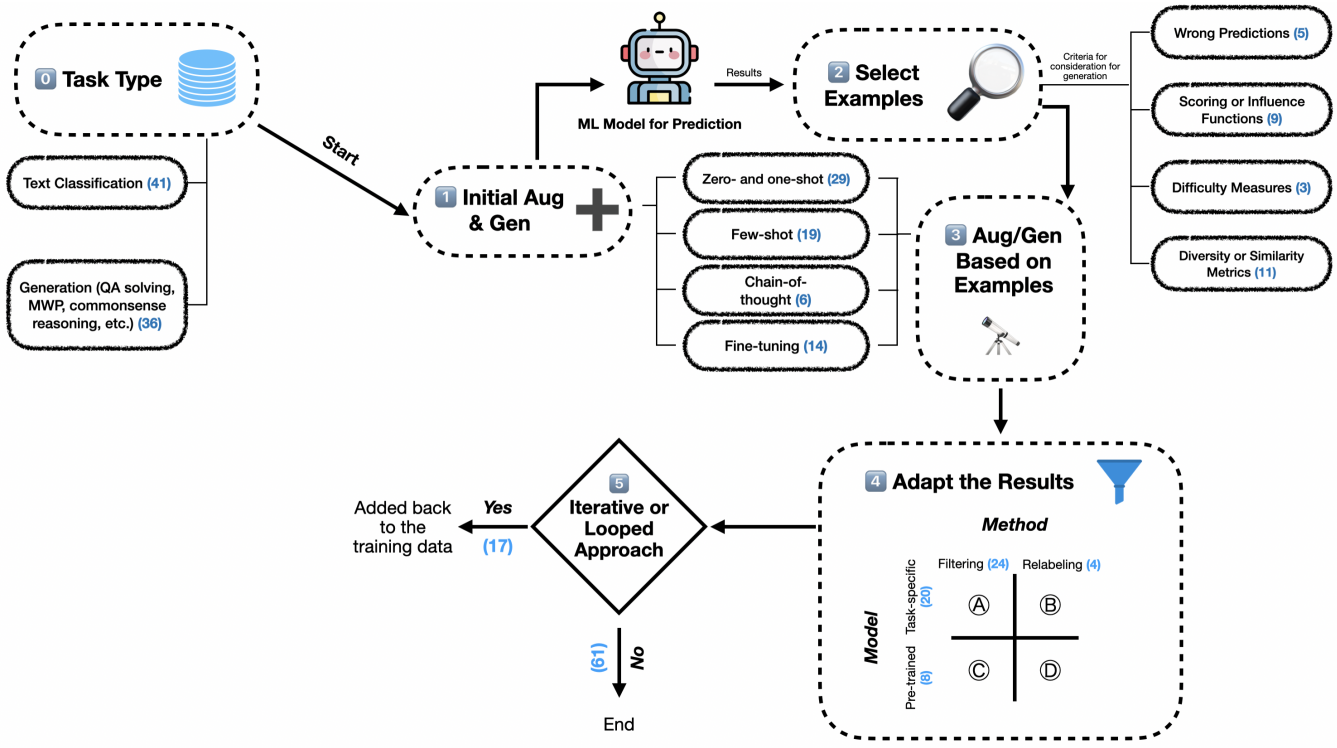

Figure 2: Our five-stage pipeline for data augmentation, extracted from our systematic literature review. The process includes task selection, initial augmentation and generation, example selection, augmentation or generation based on examples, result adaptation, and an optional iterative approach. The count of papers for each stage is indicated in blue.

the available training data to generate synthetic examples for offensive language detection. Others [62] fine-tuned a sequence-to-sequence T5 model using a dataset with limited samples in order to generate new samples for their low-resource scenarios.

After the initial data augmentation or generation process, many prior works train an early version of the final classifier on the augmented or generated data.

2) Select Examples. This stage refers to the strategy behind smartly selecting training data points at each round as a basis for further data augmentation. The approaches discussed in prior papers for this stage consist of:

- *Wrong Predictions:* 5 papers in this stage use the incorrect predictions of the model, when evaluating it on the dataset, as a basis for further augmentation. [32] considered the wrong predictions as the *weak* points of the student model in their knowledge distillation process for solving math word problems and [33] analyzed the student model's weaknesses, and then synthesized labeled samples based on an analysis across a set of NLP tasks including classification and named entity recognition.

- *Scoring or Influence Functions:* 9 papers in this stage use a certain set of model-specific metrics to find the *influence* of data points on the outcomes of the prediction, and then use the outcomes as a heuristic for choosing which samples to pick as the basis for further augmentation. For example, [70] quantified contribution to the loss for each training point using an influence function, while [69] conducted filtering using an influence function (which considers an example as *detrimental* if using it in the training data leads to a higher generalization error).

- *Difficulty Measures:* 3 papers in this stage used content-based text measures (e.g., difficulty levels of different data samples) as a heuristic for strategic selection of data points. For example, [77] proposed a self-evolution learning method that considers the learning difficulty of samples for augmentation.

- *Diversity or Similarity Metrics:* 11 papers in this stage used certain measures, such as the cosine similarity of embeddings, to maximize the diversity of the generated data or choose close or far data samples to the wrong predictions, as a basis for choosing which samples to augment from. [7], for instance, use logit suppression and temperature sampling to diversify text generation, while [75] used the original instances as queries to extract instances in the data with the most query-related degree, with the aim to generate more discriminating samples.

3) Augmentation or Generation Based on Examples. This stage refers to the methods performed on the chosen examples selected by the strategy outlined in the second stage, in order to generate new data points. This step enables the model to learn from its *mistakes* or *shortcomings* by mitigating the weaknesses of the model (e.g., on wrong predictions) depending on the strategy selected in the second stage. Prior

works (37% of the papers in our literature review) have explored prompting strategies (e.g., zero-shot and few-shot) for this stage [32, 70].

4) Adapt the Results. Due to the chance of generating data points from the wrong class, wrong format, or from low validity measures, prior works (35% of the papers in our literature review) have considered either filtering (i.e., removing data points not matching their label) [47, 67] or relabeling (i.e., changing the label of data points to the presumably correct label) [7, 50] approaches, as described in 24 and 4 papers, respectively. To find the correct labels for filtering or relabeling, prior works used a variety of methods, including basic computations, training task-specific models, and human-in-the-loop approaches (e.g., active learning-based methods) [72], or using pre-trained models (e.g., prompting LLMs to find the correct label for a data point) [6, 14], as observed in 20 and 8 papers, respectively.

5) Iterative or Looped Approach. While many works in the literature (78% of the papers) conduct the augmentation pipeline once and then train a final model on the augmented data [72, 73], some works (22% of the papers) have also explored running the augmentation pipeline multiple times, each time adding the newly-generated data back into the training set or continuing model training on the new data. This process aims to enable the model to iteratively learn from its mistakes over time and increase its accuracy on the test set [32, 33].

3. CASE STUDY: APPLICATION ON EDUCATIONAL DATA

We used the taxonomy derived in Section 2.2 as a basis for our case study on applying state-of-the-art data augmentation methods to educational datasets to improve performance in text classification tasks. We employed a step-by-step experimental design, increasingly adding additional phases (see Fig. 2) to the data augmentation process, and evaluated the performance of the resulting classifier on three different educational datasets.

3.1 Experimental Design

Our experiments focused on multi-class text classification. Before describing them in detail, we first formalize the problem. We consider T to be the set of texts $t \in T$ in our datasets. Each text is split into a set of sentences $s \in S$, where each sentence is assigned a label l. This label comes from a set of labels $L = \{l_1, l_2, l_3, \cdots\}$, which describe the category (or class) of the sentence (e.g., for a reflective text: Description, Feelings, Evaluation, Analysis, Conclusion, and Action Plan). We aim to train a sentence-based multi-class classifier C that, given a sentence s, outputs $C(s)$ as a prediction of the correct label l of sentence s.

We conducted a set of step-by-step data augmentation experiments, following the phases of our developed taxonomy. In each step, we selected the best approach for that phase and then employed it for all the following steps. In all experiments, we fine-tuned BERT text classification models [12] as the underlying classifiers. We used BERT, due to its training efficiency, relatively low memory requirements, and its common usage in educational tools [55, 30, 21, 58]. The BERT models were fine-tuned from the base model in three epochs and with the default learning rate of the `simpletransformers` library. We used 5-fold cross-validation to split the data into training and testing subsets for each experiment. We used GPT-4o (the 2024-08-06 checkpoint) for data generation[3]. The experiments were conducted over the course of three weeks, on two machines, one with an NVIDIA V100 GPU with 32 GB of memory, and another with an Apple M4 Pro system-on-a-chip with 48 GB of memory.

We evaluated five different experimental settings in total:

In the **Baseline** experiment, no data augmentation was conducted.

In the **Step-1** experiment, we performed only an initial augmentation, in which we augmented the data by generating examples from the minority classes until the dataset is balanced, experimenting with four different approaches (see Fig. 2):

- *One-shot:* We prompted the LLM with a randomly selected example sentence from a given class label $l \in L$ and then asked the LLM to come up with a sentence addressing the same topic. Specifically, we asked the LLM to use different names, words, and terminologies in its output, but keep the overall meaning and content the same. We also included a description of the different labels $l \in L$ in our system prompt[4].

- *Few-shot:* We prompted the LLM with five example sentences of each label $l \in L$ selected at random, and asked it to come up with a new sentence from the same label l. We continued this process until the dataset became balanced. We included the same system prompt as in the one-shot setting.

- *Chain-of-thought:* Following the idea provided in [65] to utilize the chain-of-thought capabilities of LLMs, we provided a similar prompt as in the one-shot approach, but asked the LLM to first think step-by-step, and then, in the last line of its response (after putting a line break), write the generated sentence with the required label $l \in L$.

- *Fine-tuning:* We fine-tuned GPT-4o following the SFT method using the API provided by OpenAI to be able to generate sentences of any specific class label l, given the natural language name of the class label.

In our next experimental setting, **Steps-1-3**, we performed an initial augmentation, selected examples, and augmented based on the selected examples[5]. We experimented with two approaches for the example selection part (step 2 of the pipeline):

[3]All prompts used for GPT-4o can be found in our GitHub repository.
[4]We did not use a pure zero-shot prompt in our experiment, as our initial exploration revealed that the LLM can not reliably produce correct sentences of the given classes in a zero-shot setting.
[5]Steps 2 and 3 had to be added together to the list of step-by-step experiments, as none can be conducted without the other.

Table 1: Components of the Gibbs reflective cycle from [44], with the number of respective sentences in our German (GRD) and English (ERD) reflective writing datasets.

Class	Description	# sentences GRD	# sentences ERD
Description	This section includes a presentation of the event the learner is reflecting on.	771 (58.9%)	375 (44.9%)
Feelings	This section includes any feelings the learners had before, at the time of, and after the event, as well as their thoughts when they were in the situation.	163 (12.4%)	121 (14.5%)
Evaluation	This section includes an honest opinion on the positive or negative points of the response the learner provided at the time of the event.	109 (8.3%)	92 (11.0%)
Analysis	This section includes possible reasons for the points mentioned in the Evaluation section. Learners may refer to references supporting the provided causes and include them in their writing in this section.	64 (4.9%)	51 (6.10%)
Conclusion	This section aims to summarize what happened and what the learner had gained from the event.	122 (9.3%)	127 (15.2%)
Action Plan	This section includes opinions on what the learner would do differently the next time they are faced with a similar situation.	81 (6.2%)	70 (8.4%)

- *Wrong Predictions:* We evaluated the fine-tuned model on the training data and identified the entries that the model had predicted incorrectly as the basis for further augmentation. The wrong predictions were added as in-context learning examples (in the case of few-shot prompts in step 3) or as a basis for paraphrasing (in the case of one-shot or chain-of-thought prompts in step 3).

- *Similarity Metrics:* We calculated the cosine similarity between embeddings of the wrongly predicted text entries and the entries of the training set, identifying the most similar and least similar data points. We then considered the most and least similar data points (as in-context examples or basis for paraphrasing, depending on the type of prompt used in step 3). We followed this approach, as adding the most similar data points can help the model learn better from a set of examples similar to the incorrect prediction, while adding the least similar data points can improve the diversity of the model inputs and training data.

For the third stage of the pipeline (augmentation or generation based on examples), we used the best-performing method (among one-shot, few-shot, and chain-of-thought) from the **Step-1** experiment. However, different from step 1 of the pipeline, we did *not* include fine-tuning in our experiments for step 3, because in our datasets, the training data at each iteration (i.e., the small number of selected examples) would have become too small for effective fine-tuning.

In the **Steps-1-4** experimental setting, we performed an initial augmentation, selected examples, augmented based on the selected examples, and adapted the results. We experimented with two approaches for adapting the results:

- *Filtering* using a *pre-trained* model (referring to part C of step 4 in Fig. 2): We prompted the LLM to specify the class label $l \in L$ that each input sentence belonged to. The model was instructed to first think step-by-step, and then write the name of the identified component, and nothing else, in the last line of its response. We filtered out any LLM-generated data point for which the label used for generation was different than the label identified by the LLM at this stage.

- *Relabeling* using a pre-trained model (referring to part D of step 4 in Fig. 2): We used the same prompt as for filtering, but instead of *removing* the sentences with the mismatching labels, we *updated* their labels.

In a final experiment, we included **Steps-1-5** of the pipeline. We selected the best-performing strategies from the prior steps and continued running the loop for five iterations to see the effects of the continuous error correction of the model on the text classification accuracy. After each iteration, we added the recently generated data back into the training dataset, trained a BERT model on the updated dataset, and continued running our pipeline from step 2.

3.2 Educational Datasets

We performed experiments on three different educational datasets, covering different educational tasks and languages. All three datasets consist of student writings (reflective writing, persuasive essays), contain sentence-based ground truth labels (e.g., claim, counterclaim, etc., for persuasive essays), enabling multi-class classification, and are imbalanced.

3.2.1 Reflective Writing

Reflective writing (i.e., journaling) refers to the process in which people write about their experiences, emotions, be-

liefs, and insights related to events that have happened in their studies or workplaces [66]. Previous works have shown the positive role of reflective writing in helping to improve the metacognitive skills of students and enhance their learning gains [45, 9]. Prior researchers in learning sciences have explored different reflective writing *frameworks* to systematically guide learners in reflecting on their experiences. One example of such frameworks is the Gibbs reflective cycle [17], which classifies a reflective text into six components in a cycle (see Table 1).

In our experiments, we used two datasets of reflective writings with sentences annotated using the Gibbs reflective cycle:

German Texts (GRD). This dataset of reflective writings [44] was collected from 60 vocational students[6] (53 identified as females, 6 as males, and 1 as non-binary, average age = 23.62, SD = 6.68), doing their apprenticeship in the domain of nursing and caring. The students used an educational writing assistant to learn reflective writing by following the steps of the Gibbs reflective cycle [17] and were then instructed to write a reflective writing essay about a past experience, focusing on a particular experience they encountered while caring for patients at their nursing center during their practice sessions. All students gave informed consent to participate in the experiment, and the study was approved by the university's ethics review board (Nr. HREC000572 and HREC 013-2021). To annotate the data using the Gibbs reflective cycle, two researchers annotated 15 reflective writings, resulting in an inter-rater agreement of $\kappa = 0.61$, indicating a *substantial* agreement [37]. After resolving the disagreements collaboratively, the researchers then independently annotated the remaining texts.

English Texts (ERD). This dataset of reflective writings [43] was collected from 100 users (70 identified as females, 29 as males, and 1 as others, average age = 25.08, SD = 2.67) from Prolific among users having a degree in health and welfare (e.g., medicine or nursing), and having a high school diploma or above. Again, all participants gave informed consent by participating in our study on Prolific, and the study was approved by the university's ethics board review (Nr. HREC000572 and HREC 013-2021). The users were instructed to use an English-language reflective writing assistant[7], which helped them learn how to write reflective texts by following the Gibbs reflective cycle. The users were asked to reflect on a situation in the workplace when things did not go as planned, and write the corresponding reflective diary. Similar to the German reflective dataset, each sentence was annotated with one of the six Gibbs reflective cycle classes by one of our trained annotators.

3.2.2 Persuasive Essays

Argumentative and persuasive writing is considered a crucial part of daily communication and decision-making [23, 53]. However, learners often lack personalized feedback on their argumentation learning process in large-scale lectures [60]. As a result, the field of argumentation mining (AM) has emerged, identifying components of an argumentative document that play a role in forming the overall argumentation chain [59].

Building upon the importance of considering argumentation mining datasets in implementing tools for persuasive writing support, in this work, we used a public dataset of persuasive essays (PED) to measure the generalizability of our findings in a different educational task and provide replicability of our results. We used the dataset of essays[8] obtained from [11], containing 325′347 sentences from a set of essays annotated for discourse elements. Due to the large size of the dataset and to be able to measure the usefulness of our pipeline and models in low-resourced settings, we only used a subset of the data to train our models. We picked 100 full texts randomly from the dataset and kept those with more than two sentences from the minority *counterclaim* and *rebuttal* classes. The classification labels of this dataset are listed in Table 2.

3.3 Experimental Evaluation (RQ2)

With our experiments, we aimed to evaluate the suitability of the state-of-the-art data augmentation approaches for educational contexts (RQ2).

3.3.1 German Reflective Writing Data

Figure 3 illustrates the balanced accuracies (BAC) for our experiments on the German Reflective Writing Dataset. We observed an overall increase in BAC from 0.55 to 0.61 by following the different stages of our pipeline. The per-class performance (in terms of F1 score for each class) can be found on our GitHub repository.

For experiment **Step-1**, we observe that all approaches except fine-tuning (BAC: 0.59) lead to balanced accuracy scores less than the baseline (BAC: 0.55, One-Shot BAC: 0.50, Few-Shot BAC: 0.55, CoT BAC: 0.49). When investigating the per-class accuracies, we found that the improved BAC is mainly due to an improved classifier performance on the minority class *Analysis* (Baseline F1: 0.11, Fine-Tuning F1: 0.20).

For the **Steps-1-3** experiment, we found very limited differences between using wrong predictions (BAC: 0.60) and using similarity metrics (BAC: 0.59) for example selection.

In the **Steps-1-4** experiment, we found slightly higher effectiveness of filtering data (BAC: 0.61) compared to relabeling (BAC: 0.58), which was especially pronounced in the class-specific F1 scores for a subset of classes (*Feelings*: F1 Filtering = 0.72, F1 Relabeling = 0.65; *Analysis*: F1 Filtering = 0.26, F1 Relabeling = 0.18).

Finally, when running the data augmentation pipeline iteratively (**Steps-1-5** experiment), we received scores very similar to the previous experiment with no iterations (second iteration BAC: 0.604, third iteration BAC: 0.603, fourth iteration BAC: 0.596, fifth iteration BAC: 0.601). However,

[6]Five students were excluded, as they did not correctly complete the field study, failing to answer the survey questions or submit their writing.

[7]All students followed the instructions of the system and wrote their texts in English, except one student who provided a text in Portuguese.

[8]https://www.kaggle.com/competitions/feedback-prize-2021/data

Table 2: Description of class labels along with sentence counts per label for the persuasive essays dataset (PED) [11].

Class	Description	# sentences PED
Lead	This component includes an opening that uses a statistic, quotation, description, or another technique to grab the attention of the readers and introduce the main idea.	263 (8.2%)
Position	This component includes expressing an opinion or conclusion on the central question.	123 (3.8%)
Claim	This component includes a claim that *supports* the position.	421 (13.1%)
Counterclaim	This component includes a claim that *opposes* the position or refutes another claim.	402 (12.5%)
Rebuttal	This component includes a claim that counters or disproves a counterclaim.	425 (13.2%)
Evidence	This component includes ideas or examples used to back up claims, counterclaims, rebuttals, or the position.	1213 (37.6%)
Concluding Statement	This component includes a statement that restates the claims and the position.	378 (11.7%)

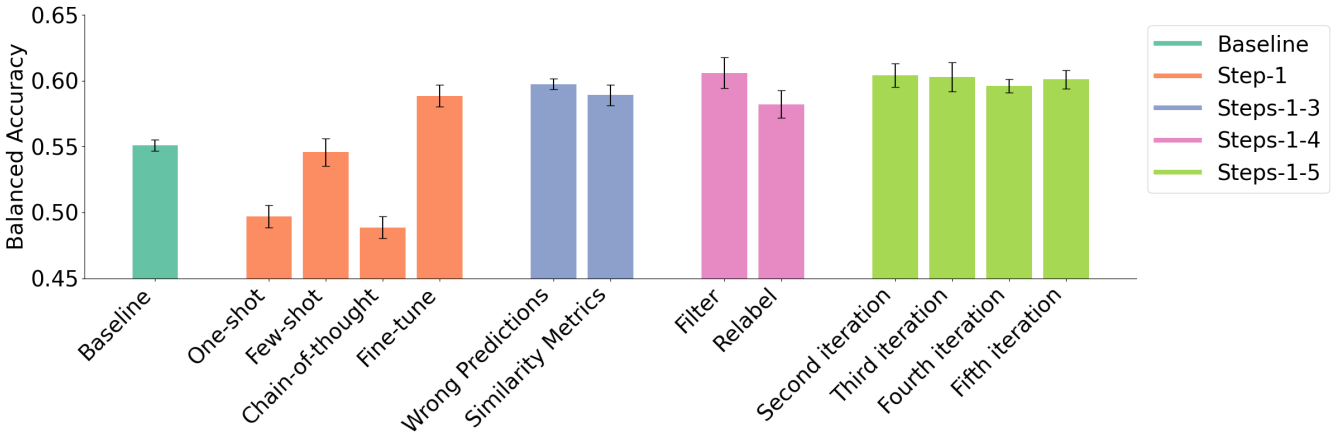

Figure 3: Balanced accuracies (with standard errors) for each experiment on the GRD.

it is important to acknowledge the noticeable increase in the accuracy metrics over the original baseline (BAC: 0.55).

3.3.2 *English Reflective Writing Data*

Figure 4 illustrates the balanced accuracies (BAC) for our experiments on the English Reflective Writing Dataset. We observed an overall increase in BAC from 0.73 to 0.77 by following the different stages of our pipeline. Again, the per-class performance (in terms of F1 score for each class) can be found on our GitHub repository.

We found slight increases in overall BAC for all of the initial data augmentation approaches (**Step-1** experiment) compared to the baseline (BAC: 0.73), with the effect being most pronounced for chain-of-thought prompting (BAC: 0.77) and fine-tuning (BAC: 0.77), and least pronounced for one-shot prompting (BAC: 0.73).

Interestingly, in our **Steps-1-3** experiment, we found that introducing the example selection and augmentation process (steps 2 and 3) was harmful to the model performance (BAC Wrong Predictions: 0.73, BAC Similarity Metrics: 0.74), which was especially reflected in lower F1 on specific classes (Analysis: F1 CoT = 0.56, F1 Wrong Predictions = 0.52, F1 Similarity Metrics = 0.52; Action Plan: F1 CoT = 0.92, F1 Wrong Predictions = 0.88, F1 Similarity Metrics = 0.89). These results contrast with our findings on the German Reflective Writing Dataset, where steps 2-3 led to slight improvements.

For the next step in the pipeline (**Steps-1-4** experiment), we observed similar accuracies for filtering (BAC: 0.74) and re-labeling (BAC: 0.75). At a per-class view, relabeling achieved better F1 scores for *Feelings* (F1 Filtering: 0.87, F1 Relabeling: 0.89) and *Evaluation* (F1 Filtering: 0.46, F1 Relabeling: 0.49), while the opposite is true for *Analysis* (F1 Filtering: 0.53, F1 Relabeling: 0.51), and *Conclusion* (F1 Filtering: 0.80, F1 Relabeling: 0.78).

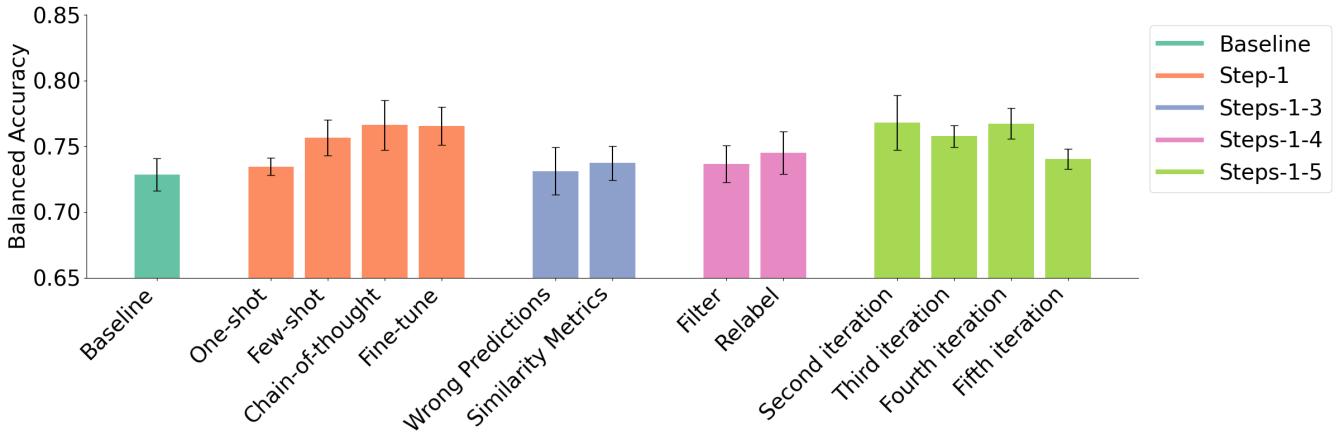

Figure 4: Balanced accuracies (with standard errors) for each experiment on the ERD.

Finally, using an iterative approach demonstrated performance increases, where using two iterations (BAC: 0.77) achieved the overall best results across all experiments on the dataset. However, the further iterations did not consistently improve the overall balanced accuracy (BAC third iteration: 0.76, BAC fourth iteration: 0.77, BAC fifth iteration: 0.74).

3.3.3 Persuasive Essays Dataset

The balanced accuracies (including standard errors) achieved in the experiments on the Persuasive Essay Dataset are illustrated in Fig. 5). We observed an overall increase in BAC from 0.44 to 0.48 by following the different stages of our pipeline. We again report the full per-class results on our GitHub repository.

We observed that all initial data augmentation approaches (**Steps-1** experiment) achieved a higher BAC than the baseline (BAC: 0.44), but that one-shot (BAC: 0.47), few-shot (BAC: 0.47), CoT prompting (BAC: 0.47), and fine-tuning (BAC: 0.47) achieved similar results.

Similar to the English Reflective Writing Dataset, we observed that introducing example selection and example-based data augmentation (**Steps-1-3** experiment) led to lower model performance (BAC Wrong Predictions: 0.45, BAC Similarity Metrics: 0.45) than merely using initial data augmentation.

For our **Steps-1-4** experiment, we found slightly higher effectiveness of filtering data (BAC: 0.48) compared to relabeling (BAC: 0.45), especially pronounced in the F1 scores for specific classes, including *Claim* (F1 Filtering: 0.48, F1 Relabeling: 0.44), *Lead* (F1 Filtering: 0.40, F1 Relabeling: 0.34), and *Position* (F1 Filtering: 0.44, F1 Relabeling: 0.39). This result is in line with the findings for the same experiment on the German Reflective Writing Dataset.

Finally, when employing an iterative approach (**Steps-1-5** experiment), we found that while the BAC increased with every subsequent iteration (BAC second iteration; 0.46, BAC third iteration: 0.47, BAC fourth iteration: 0.47, BAC fifth iteration: 0.48), it remained relatively lower than the BAC achieved with filtering in the **Steps-1-4** experiment.

4. DISCUSSION

Using intelligent and interactive systems, which include NLP models in their backbone, has been shown to be beneficial in supporting students in the context of ITS and educational writing assistants. However, NLP models, e.g., transformer-based classifiers, need an extensive amount of balanced or near-balanced data for optimal performance, a characteristic commonly lacking in many educational datasets. To move in the direction of addressing this issue, in this paper, we first conducted a systematic literature review on recent LLM-based approaches to data augmentation, resulting in a taxonomy reflecting the data augmentation pipeline. We found certain stages of our pipeline to be covered in more detail among prior works; for example, zero- or one-shot prompting (29 papers, 37% of all), few-shot prompting (19 papers, 24% of all), and fine-tuning models to generate new data points (14 papers, 18% of all) were used extensively in the data augmentation and generation literature. Moreover, a notable number of papers (24 papers, 31% of all) consider filtering the generated entries to ensure a high validity and expressiveness of the training data. However, certain approaches in the pipeline were less explored in the literature, necessitating further investigation. For example, we found only a limited number of papers that used CoT prompts (6 papers, 8% of all), even though research shows the added benefits of chain-of-thought prompting on model performance [65]. Moreover, certain methods for selecting examples, e.g., using difficulty measures (3 papers, 4% of all), were only scarcely covered in the papers from our literature review. Finally, only a few papers conducted a relabeling process (4 papers, 5% of all) after generating new data points. This suggests the need for future work on best practices of relabeling data points using pre-trained or task-specific language models.

Our pipeline serves as a conceptual framework for future researchers focusing on data augmentation and generation methods, by allowing them to systematically follow the common steps and strategies used for data augmentation and generation, and extracting the underexplored approaches in each stage that could potentially turn out to be use-

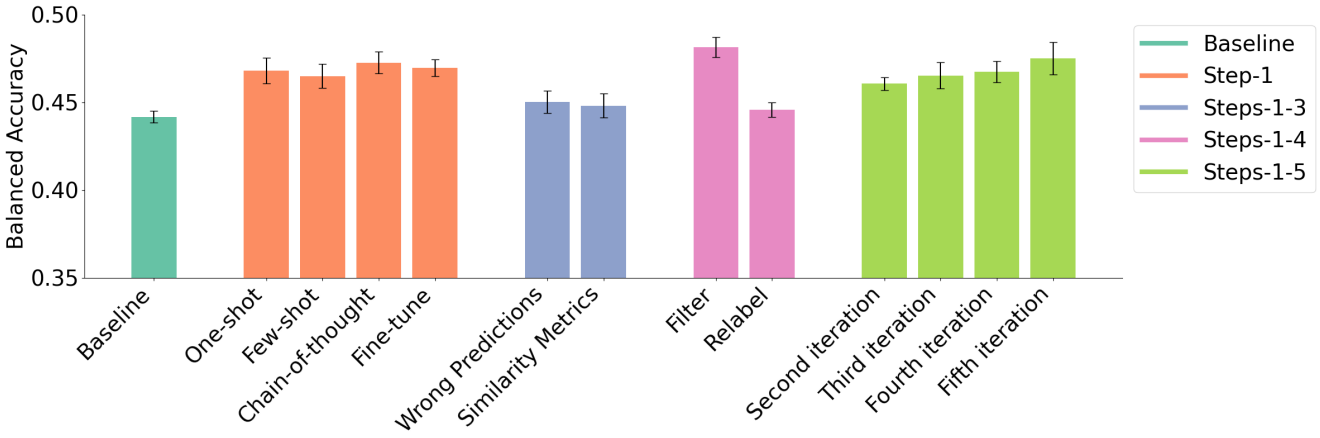

Figure 5: Balanced accuracies (with standard errors) for each experiment on the PED.

ful for certain downstream tasks. We applied the stages of our pipeline sequentially to three imbalanced educational datasets of reflective and persuasive writings. Particularly, regarding the reflective writing datasets, collecting and annotating our own datasets in a rigorous process ensured that any issue in the generalization of models (e.g., low performance in certain classes after multiple training epochs) did not likely stem from a low annotation quality, but from the nature of the data itself. Generally, we found that applying data augmentation along the five stages led to final improvements in overall balanced accuracy, as well as increases in F1 scores of certain underrepresented classes (e.g., *Analysis* for the reflective writing datasets, and *Lead* or *Position* for the persuasive essays dataset), confirming the general applicability of LLM-driven data augmentation methods across a set of educational datasets. However, we observed that not all augmentation approaches led to an improved accuracy when individually considered. For example, using similarity metrics in stage 2 was harmful to the overall balanced accuracy for the English reflective writings and persuasive essays datasets. Moreover, continuing the iterative augmentation approach for more iterations was not helpful in terms of overall balanced accuracy on the reflective datasets and did not lead to consistent improvements in minority classes.

We believe that the differences between our results and those of prior works considering each stage of our pipeline individually could either come from the differences in tasks and data or the adaptations that we had to undertake in order to use a similar method for our text classification task. Nevertheless, the overall gains in balanced accuracy and per-class F1 scores of minority classes indicate the usefulness of our pipeline for improving classification accuracy on educational datasets. All in all, our results suggest that the state-of-the-art approaches to data augmentation can be useful for improving classification accuracy. This is signified by our provided pipeline, which provides a systematic way of approaching the task of data augmentation or generation for a variety of downstream tasks.

Our work comes with several limitations. First, we only explored the applicability of a subset of methods from our pipeline on three educational datasets. Future work should therefore apply the methods to more tasks, datasets, and models (e.g., RoBERTa, DeBERTa, etc.) to ensure generalizability. Second, there is a possibility that any difference in the implementation of the stages of our pipeline, e.g., the prompts we used for the LLM-driven approaches, the configuration of our fine-tuned models, or the details of calculating similarity metrics, can lead to a variety of results with possibly different interpretations on a variety of datasets. Third, we only employed one LLM, namely GPT-4o, as part of our augmentation pipeline. While GPT-4o has shown promising performance across a variety of educational [38, 68] and non-educational [28, 76] tasks, it remains a proprietary model. We performed initial experiments with the small open-source Llama 3.1 8B model, resulting in subpar performance compared to GPT-4o (e.g., a mean balanced accuracy of 0.55 after fine-tuning the model using LoRA on the German reflective writing dataset, almost the same as the baseline performance of 0.55 without any augmentation, versus a mean balanced accuracy of 0.59 obtained by fine-tuning GPT-4o). Nevertheless, future work should continue experimenting with open-source models.

5. CONCLUSION

In this paper, we presented a systematic literature review on LLM-driven data augmentation for text classification, with the goal of improving the classification accuracy of the models trained on educational datasets. We proposed a five-stage pipeline based on the insights extracted from our literature review. Our empirical experiments on three educational datasets, including different tasks and languages, demonstrated the effectiveness of LLM-driven augmentation techniques in addressing class imbalance and improving classifier performance. Our findings highlight the promise of LLMs in enhancing the classification accuracy of models trained on educational data, while also emphasizing the need for further research on optimizing augmentation strategies for domain-specific educational contexts.

Ethical Considerations

Our study on LLM-driven data augmentation for text classification on educational datasets naturally necessitates careful attention to ethical considerations, particularly regarding

data privacy, bias mitigation, and the responsible use of synthetic data. Since we leveraged real student-generated texts for augmentation, we ensured following a data anonymization process to protect student identities and prevent unintended exposure of sensitive information [48]. However, given that LLMs may propagate biases present in their training data, we suggest future work to critically evaluate the generated texts quantitatively and qualitatively, in order to minimize skewed representations that could disproportionately affect certain student groups [36, 61], as well as to mitigate the possible risks of using LLMs in educational settings [3, 57]. Finally, as GPT-4o, the LLM we used, is a proprietary model, we documented our methods and dataset operations to enhance transparency and reproducibility. This highlights the need for future researchers to assess and refine LLM-driven augmentation strategies for educational applications by experimenting with open-source models.

Acknowledgments

This project was substantially financed by the Swiss State Secretariat for Education, Research, and Innovation (SERI).

6. REFERENCES

[1] K. Aggarwal, H. Jin, and A. Ahmad. ECG-QALM: Entity-controlled synthetic text generation using contextual Q&A for NER. In *Findings of the association for computational linguistics: ACL 2023*, pages 5649–5660, 2023.

[2] T. Alhindi, S. Muresan, and P. Nakov. Large language models are few-shot training example generators: a case study in fallacy recognition. *arXiv preprint arXiv:2311.09552*, 2023.

[3] B. Borges, N. Foroutan, D. Bayazit, A. Sotnikova, S. Montariol, T. Nazaretzky, M. Banaei, A. Sakhaeirad, P. Servant, S. P. Neshaei, and others. Could ChatGPT get an engineering degree? Evaluating higher education vulnerability to AI assistants. *Proceedings of the National Academy of Sciences*, 121(49):e2414955121, 2024. Publisher: National Academy of Sciences.

[4] C. Casula and S. Tonelli. Generation-based data augmentation for offensive language detection: Is it worth it? In *Proceedings of the 17th conference of the european chapter of the association for computational linguistics*, pages 3359–3377, 2023.

[5] H. Chae, Y. Song, K. T.-i. Ong, T. Kwon, M. Kim, Y. Yu, D. Lee, D. Kang, and J. Yeo. Dialogue chain-of-thought distillation for commonsense-aware conversational agents. *arXiv preprint arXiv:2310.09343*, 2023.

[6] J. Choi, Y. Kim, S. Yu, J. Yun, and Y. Kim. UniGen: Universal domain generalization for sentiment classification via zero-shot dataset generation. *arXiv preprint arXiv:2405.01022*, 2024.

[7] J. J. Y. Chung, E. Kamar, and S. Amershi. Increasing diversity while maintaining accuracy: Text data generation with large language models and human interventions. *arXiv preprint arXiv:2306.04140*, 2023.

[8] K. Cochran, C. Cohn, N. Hutchins, G. Biswas, and P. Hastings. Improving automated evaluation of formative assessments with text data augmentation. In *International conference on artificial intelligence in education*, pages 390–401. Springer, 2022.

[9] J. Colomer, T. Serra, D. Cañabate, and R. Bubnys. Reflective learning in higher education: Active methodologies for transformative practices. *Sustainability*, 2020. Issue: 9 Pages: 3827 Volume: 12.

[10] J.-P. Corbeil and H. A. Ghadivel. Bet: A backtranslation approach for easy data augmentation in transformer-based paraphrase identification context. *arXiv preprint arXiv:2009.12452*, 2020.

[11] S. A. Crossley, P. Baffour, Y. Tian, A. Picou, M. Benner, and U. Boser. The persuasive essays for rating, selecting, and understanding argumentative and discourse elements (PERSUADE) corpus 1.0. *Assessing Writing*, 54:100667, 2022. Publisher: Elsevier.

[12] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[13] F. Di Palo, P. Singhi, and B. Fadlallah. Performance-guided LLM knowledge distillation for efficient text classification at scale. *arXiv preprint arXiv:2411.05045*, 2024.

[14] H. Do and G. G. Lee. Aspect-based semantic textual similarity for educational test items. In *International conference on artificial intelligence in education*, pages 344–352. Springer, 2024.

[15] A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Yang, A. Fan, and others. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[16] K. Feng, C. Li, X. Zhang, J. Zhou, Y. Yuan, and G. Wang. Keypoint-based progressive chain-of-thought distillation for llms. *arXiv preprint arXiv:2405.16064*, 2024.

[17] G. Gibbs. Learning by doing: A guide to teaching and learning methods. *Further Education Unit*, 1988. Publisher: Oxford Polytechnic.

[18] N. T. Heffernan and K. R. Koedinger. Intelligent tutoring systems are missing the tutor: Building a more strategic dialog-based tutor. In *Building dialogue systems for tutorial applications, papers of the 2000 AAAI fall symposium*, pages 14–19. AAAI Press Menlo Park, CA, 2000.

[19] A. Hurst, A. Lerer, A. P. Goucher, A. Perelman, A. Ramesh, A. Clark, A. Ostrow, A. Welihinda, A. Hayes, A. Radford, and others. GPT-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

[20] X. Jin and H. Ji. Schema-based data augmentation for event extraction. In *Proceedings of the 2024 joint international conference on computational linguistics, language resources and evaluation (LREC-COLING 2024)*, pages 14382–14392, 2024.

[21] S. Kakarla, C. Borchers, D. Thomas, S. Bhushan, and K. R. Koedinger. Comparing few-shot prompting of GPT-4 llms with BERT classifiers for open-response assessment in tutor equity training. *arXiv preprint arXiv:2501.06658*, 2025.

[22] S. Keele and others. Guidelines for performing systematic literature reviews in software engineering. Technical report, Technical report, ver. 2.3 ebse technical report. ebse, 2007.

[23] D. Kuhn. Thinking as argument. *Harvard educational review*, 62(2):155–179, 1992. Publisher: Harvard Education Publishing Group.

[24] T. Käser and G. Alexandron. Simulated learners in educational technology: A systematic literature review and a turing-like test. *International Journal of Artificial Intelligence in Education*, 34(2):545–585, 2024. Publisher: Springer.

[25] J. H. Lee, J. S. Park, and J. G. Shon. A BERT-based automatic scoring model of korean language learners' essay. *Journal of Information Processing Systems*, 18(2):282–291, 2022. Publisher: Korea Information Processing Society.

[26] M. Lee, K. I. Gero, J. J. Y. Chung, S. B. Shum, V. Raheja, H. Shen, S. Venugopalan, T. Wambsganss, D. Zhou, E. A. Alghamdi, T. August, A. Bhat, M. Z. Choksi, S. Dutta, J. L. Guo, M. N. Hoque, Y. Kim, S. Knight, S. P. Neshaei, A. Shibani, D. Shrivastava, L. Shroff, A. Sergeyuk, J. Stark, S. Sterman, S. Wang, A. Bosselut, D. Buschek, J. C. Chang, S. Chen, M. Kreminski, J. Park, R. Pea, E. H. R. Rho, Z. Shen, and P. Siangliulue. A design space for intelligent and interactive writing assistants. In *Proceedings of the CHI conference on human factors in computing systems*, Chi '24, New York, NY, USA, 2024. Association for Computing Machinery. Number of pages: 35 Place: ¡conf-loc¿, ¡city¿Honolulu¡/city¿, ¡state¿HI¡/state¿, ¡country¿USA¡/country¿, ¡/conf-loc¿ tex.articleno: 1054.

[27] Z. Levonian, O. Henkel, C. Li, M.-E. Postle, and others. Designing safe and relevant generative chats for math learning in intelligent tutoring systems. *Journal of Educational Data Mining*, 17(1), 2025.

[28] Q. Li and P. H. Li. Transformative potential of GPT-4o in clinical immunology and allergy: Opportunities and challenges of real-time voice interaction. *Asia Pacific Allergy*, 14(4):232–233, 2024. Publisher: LWW.

[29] Y. Li and J. Yuan. Generative data augmentation with contrastive learning for zero-shot stance detection. In *Proceedings of the 2022 conference on empirical methods in natural language processing*, pages 6985–6995, 2022.

[30] Z. Li, J. Yang, J. Wang, L. Shi, and S. Stein. Integrating lstm and bert for long-sequence data analysis in intelligent tutoring systems. *arXiv preprint arXiv:2405.05136*, 2024.

[31] Z. Li, H. Zhu, Z. Lu, and M. Yin. Synthetic data generation with large language models for text classification: Potential and limitations. *arXiv preprint arXiv:2310.07849*, 2023.

[32] Z. Liang, W. Yu, T. Rajpurohit, P. Clark, X. Zhang, and A. Kalyan. Let GPT be a math tutor: Teaching math word problem solvers with customized exercise generation. In H. Bouamor, J. Pino, and K. Bali, editors, *Proceedings of the 2023 conference on empirical methods in natural language processing*, pages 14384–14396, Singapore, Dec. 2023. Association for Computational Linguistics.

[33] C. Liu, Y. Kang, F. Zhao, K. Kuang, Z. Jiang, C. Sun, and F. Wu. Evolving knowledge distillation with large language models and active learning. *arXiv preprint arXiv:2403.06414*, 2024.

[34] P. Liu, X. Wang, C. Xiang, and W. Meng. A survey of text data augmentation. In *2020 international conference on computer communication and network security (CCNS)*, pages 191–195. IEEE, 2020.

[35] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv preprint arXiv:1907.11692*, 2019.

[36] L. Lucy and D. Bamman. Gender and representation bias in GPT-3 generated stories. In *Proceedings of the third workshop on narrative understanding*, pages 48–55, 2021.

[37] M. L. McHugh. Interrater reliability: the kappa statistic. *Biochemia medica*, 22(3):276–282, 2012. Publisher: Medicinska naklada.

[38] H. Moon, R. Davis, S. P. Neshaei, and P. Dillenbourg. Using large multimodal models to extract knowledge components for knowledge tracing from multimedia question information. *arXiv preprint arXiv:2409.20167*, 2024.

[39] M. Munikar, S. Shakya, and A. Shrestha. Fine-grained sentiment classification using BERT. In *2019 artificial intelligence for transforming business and society (AITB)*, volume 1, pages 1–5. IEEE, 2019.

[40] M. Y. Mustafa, A. Tlili, G. Lampropoulos, R. Huang, P. Jandrić, J. Zhao, S. Salha, L. Xu, S. Panda, S. López-Pernas, and others. A systematic review of literature reviews on artificial intelligence in education (AIED): a roadmap to a future research agenda. *Smart Learning Environments*, 11(1):1–33, 2024. Publisher: Springer.

[41] J. Nehyba and M. Štefánik. Applications of deep language models for reflective writings. *Education and Information Technologies*, 28(3):2961–2999, 2023. Publisher: Springer.

[42] S. P. Neshaei, R. L. Davis, A. Hazimeh, B. Lazarevski, P. Dillenbourg, and T. Käser. Towards modeling learner performance with large language models. In B. Paaßen and C. D. Epp, editors, *Proceedings of the 17th international conference on educational data mining*, pages 759–768, Atlanta, Georgia, USA, July 2024. International Educational Data Mining Society.

[43] S. P. Neshaei, P. Mejia-Domenzain, R. L. Davis, and T. Käser. Metacognition meets AI: Empowering reflective writing with large language models. *British Journal of Educational Technology*, 2025. Publisher: Wiley Online Library.

[44] S. P. Neshaei, T. Wambsganss, H. El Bouchrifi, and T. Käser. MindMate: Exploring the effect of conversational agents on reflective writing. In *Proceedings of the extended abstracts of the CHI conference on human factors in computing systems*, pages 1–9, 2025.

[45] V. D. O'Loughlin and L. M. Griffith. Developing student metacognition through reflective writing in an upper level undergraduate anatomy course. *Anatomical Sciences Education*, 13(6):680–693, 2020. Publisher: Wiley Online Library.

[46] L. Paletto, V. Basile, and R. Esposito. Label augmentation for zero-shot hierarchical text classification. In *Proceedings of the 62nd annual*

meeting of the association for computational linguistics (volume 1: Long papers), pages 7697–7706, 2024.

[47] P. Patwa, S. Filice, Z. Chen, G. Castellucci, O. Rokhlenko, and S. Malmasi. Enhancing low-resource LLMs classification with PEFT and synthetic data. *arXiv preprint arXiv:2404.02422*, 2024.

[48] I. Potgieter. Privacy concerns in educational data mining and learning analytics. *The International Review of Information Ethics*, 28, 2020.

[49] E. Prihar, M. Lee, M. Hopman, A. T. Kalai, S. Vempala, A. Wang, G. Wickline, A. Murray, and N. Heffernan. Comparing different approaches to generating mathematics explanations using large language models. In *International conference on artificial intelligence in education*, pages 290–295. Springer, 2023.

[50] G. Sahu, O. Vechtomova, D. Bahdanau, and I. Laradji. PromptMix: a class boundary augmentation method for large language model distillation. In H. Bouamor, J. Pino, and K. Bali, editors, *Proceedings of the 2023 conference on empirical methods in natural language processing*, pages 5316–5327, Singapore, Dec. 2023. Association for Computational Linguistics.

[51] V. Sanh. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

[52] N. Scaria, S. Dharani Chenna, and D. Subramani. Automated educational question generation at different bloom's skill levels using large language models: Strategies and evaluation. In *International conference on artificial intelligence in education*, pages 165–179. Springer, 2024.

[53] O. Scheuer, F. Loll, N. Pinkwart, and B. M. McLaren. Computer-supported argumentation: A review of the state of the art. *International Journal of Computer-supported collaborative learning*, 5:43–102, 2010. Publisher: Springer.

[54] B. A. Schwendimann, M. J. Rodriguez-Triana, A. Vozniuk, L. P. Prieto, M. S. Boroujeni, A. Holzer, D. Gillet, and P. Dillenbourg. Perceiving learning at a glance: A systematic literature review of learning dashboard research. *IEEE transactions on learning technologies*, 10(1):30–41, 2016. Publisher: IEEE.

[55] J. T. Shen, M. Yamashita, E. Prihar, N. Heffernan, X. Wu, B. Graff, and D. Lee. Mathbert: A pre-trained language model for general nlp tasks in mathematics education. *arXiv preprint arXiv:2106.07340*, 2021.

[56] Y. Shi, T. ValizadehAslani, J. Wang, P. Ren, Y. Zhang, M. Hu, L. Zhao, and H. Liang. Improving imbalanced learning by pre-finetuning with data augmentation. In *Fourth international workshop on learning with imbalanced domains: Theory and applications*, pages 68–82. PMLR, 2022.

[57] M. M. Van Wyk. Is ChatGPT an opportunity or a threat? Preventive strategies employed by academics related to a GenAI-based LLM at a faculty of education. *Journal of applied learning and teaching*, 7(1), 2024.

[58] T. Wambsganss, T. Kueng, M. Soellner, and J. M. Leimeister. ArgueTutor: An adaptive dialog-based learning system for argumentation skills. In *Proceedings of the 2021 CHI conference on human factors in computing systems*, pages 1–13, 2021.

[59] T. Wambsganss and C. Niklaus. Modeling persuasive discourse to adaptively support students' argumentative writing. In *Proceedings of the 60th annual meeting of the association for computational linguistics (volume 1: Long papers)*, pages 8748–8760, 2022.

[60] T. Wambsganss, C. Niklaus, M. Cetto, M. Söllner, S. Handschuh, and J. M. Leimeister. AL: An adaptive learning support system for argumentation skills. In *Proceedings of the 2020 CHI conference on human factors in computing systems*, pages 1–14, 2020.

[61] T. Wambsganss, X. Su, V. Swamy, S. P. Neshaei, R. Rietsche, and T. Käser. Unraveling downstream gender bias from large language models: A study on AI educational writing assistance. *arXiv preprint arXiv:2311.03311*, 2023.

[62] C. Wang, G. F. Pontiveros, S. Derby, and T. K. Wijaya. STA: Self-controlled text augmentation for improving text classifications. *arXiv preprint arXiv:2302.12784*, 2023.

[63] F. Weber, T. Wambsganss, S. P. Neshaei, and M. Soellner. Structured persuasive writing support in legal education: A model and tool for German legal case solutions. In *Findings of the association for computational linguistics: ACL 2023*, pages 2296–2313, 2023.

[64] J. Wei, M. Bosma, V. Y. Zhao, K. Guu, A. W. Yu, B. Lester, N. Du, A. M. Dai, and Q. V. Le. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*, 2021.

[65] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, and others. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[66] K. Williams, M. Woolliams, and J. Spiro. *Reflective writing*. Bloomsbury Publishing, 2020.

[67] R. Xiao, Y. Dong, J. Zhao, R. Wu, M. Lin, G. Chen, and H. Wang. Freeal: Towards human-free active learning in the era of large language models. *arXiv preprint arXiv:2311.15614*, 2023.

[68] W. Xing, T. Zhu, J. Wang, and B. Liu. A survey on MLLMs in education: Application and future directions. *Future Internet*, 2024. Publisher: MDPI.

[69] Y. Yang, C. Malaviya, J. Fernandez, S. Swayamdipta, R. Le Bras, J.-P. Wang, C. Bhagavatula, Y. Choi, and D. Downey. Generative data augmentation for commonsense reasoning. In T. Cohn, Y. He, and Y. Liu, editors, *Findings of the association for computational linguistics: EMNLP 2020*, pages 1008–1025, Online, Nov. 2020. Association for Computational Linguistics.

[70] J. Ye, J. Gao, Z. Wu, J. Feng, T. Yu, and L. Kong. ProGen: Progressive zero-shot dataset generation via in-context feedback. In Y. Goldberg, Z. Kozareva, and Y. Zhang, editors, *Findings of the association for computational linguistics: EMNLP 2022*, pages 3671–3683, Abu Dhabi, United Arab Emirates, Dec. 2022. Association for Computational Linguistics.

[71] Z.-X. Yong, C. Menghini, and S. H. Bach. LexC-gen: Generating data for extremely low-resource languages

with large language models and bilingual lexicons. *arXiv preprint arXiv:2402.14086*, 2024.

[72] W. You, S. Yin, X. Zhao, Z. Ji, G. Zhong, and J. Bai. MuMath: Multi-perspective data augmentation for mathematical reasoning in large language models. In *Findings of the association for computational linguistics: NAACL 2024*, pages 2932–2958, 2024.

[73] L. Zeng. Leveraging large language models for code-mixed data augmentation in sentiment analysis. *arXiv preprint arXiv:2411.00691*, 2024.

[74] H. Zhang, H. Liang, L. Zhan, A. Lam, and X.-M. Wu. Revisit few-shot intent classification with PLMs: Direct fine-tuning vs. continual pre-training. *arXiv preprint arXiv:2306.05278*, 2023.

[75] J. Zhang, H. Gao, P. Zhang, B. Feng, W. Deng, and Y. Hou. LA-UCL: LLM-augmented unsupervised contrastive learning framework for few-shot text classification. In *Proceedings of the 2024 joint international conference on computational linguistics, language resources and evaluation (LREC-COLING 2024)*, pages 10198–10207, 2024.

[76] N. Zhang, Z. Sun, Y. Xie, H. Wu, and C. Li. The latest version ChatGPT powered by GPT-4o: what will it bring to the medical field? *International Journal of Surgery*, pages 10–1097, 2024. Publisher: LWW.

[77] H. Zheng, Q. Zhong, L. Ding, Z. Tian, X. Niu, C. Wang, D. Li, and D. Tao. Self-evolution learning for mixup: Enhance data augmentation on few-shot text classification tasks. In H. Bouamor, J. Pino, and K. Bali, editors, *Proceedings of the 2023 conference on empirical methods in natural language processing*, pages 8964–8974, Singapore, Dec. 2023. Association for Computational Linguistics.

[78] T. Zou, Y. Liu, P. Li, J. Zhang, J. Liu, and Y.-Q. Zhang. FuseGen: PLM fusion for data-generation based zero-shot learning. *arXiv preprint arXiv:2406.12527*, 2024.