Generalizable and Efficient Automated Scoring with a Knowledge-Distilled Multi-Task Mixture-of-Experts

Luyang Fang[1,2], **Tao Wang**[1], **Ping Ma**[1], **Xiaoming Zhai**[2*]

[1]Department of Statistics, University of Georgia
[2]AI4STEM Education Center, University of Georgia
Luyang.Fang@uga.edu, Tao.Wang2@uga.edu, pingma@uga.edu, Xiaoming.Zhai@uga.edu

Abstract

Automated scoring of written constructed responses typically relies on separate models per task, straining computational resources, storage, and maintenance in real-world education settings. We propose UniMoE-Guided, a knowledge-distilled multi-task Mixture-of-Experts (MoE) approach that transfers expertise from multiple task-specific large models (teachers) into a single compact, deployable model (student). The student combines (i) a shared encoder for cross-task representations, (ii) a gated MoE block that balances shared and task-specific processing, and (iii) lightweight task heads. Trained with both ground-truth labels and teacher guidance, the student matches strong task-specific models while being far more efficient to train, store, and deploy. Beyond efficiency, the MoE layer improves transfer and generalization: experts develop reusable skills that boost cross-task performance and enable rapid adaptation to new tasks with minimal additions and tuning. On nine NGSS-aligned science-reasoning tasks (seven for training/evaluation and two held out for adaptation), UniMoE-Guided attains performance comparable to per-task models while using $\sim 6\times$ less storage than maintaining separate students, and $87\times$ less than the 20B-parameter teacher. The method offers a practical path toward scalable, reliable, and resource-efficient automated scoring for classroom and large-scale assessment systems.

Code — https://github.com/LuyangFang/UniMoE

Introduction

Automated scoring systems have become indispensable in modern educational assessment by enabling efficient evaluation of students' open-ended responses, particularly in science and STEM domains (Page 1966; Dikli 2006; Moharreri, Ha, and Nehm 2014; Jescovitch et al. 2021). As curriculum frameworks like the Next Generation Science Standards (NGSS) promote complex performance tasks to assess multidimensional understanding (Harris, Krajcik, and Pellegrino 2024), demand for reliable, scalable automated scoring grows. Transformer-based deep learning models, including large language models (LLMs), deliver strong accuracy for automated essay scoring (AES) and short-answer grading (ASAG), enhancing measurement fidelity and operational efficiency (Zhai, He, and Krajcik 2022; Lee et al. 2024; Latif et al. 2024a). However, the dominant cost driver remains the maintenance and deployment of numerous per-item or per-task models. In classroom assessments, it is standard practice to train models per task to ensure alignment with item-specific rubrics and expert criteria. This proliferation of models generates many artifacts requiring storage, monitoring, and redeployment; the burden is especially severe in classroom assessment systems and statewide or district-scale testing programs where storage constraints and inference speed are critical (Liu et al. 2021; Zhai, He, and Krajcik 2022; Reidy et al. 2023).

Methods like parameter-efficient tuning (Han et al. 2024; Mahmoud, Nabil, and Torki 2024) reduce the number of trainable parameters and update size. For example, LoRA often matches full fine-tuning without added inference latency by inserting low-rank matrices while freezing the backbone (Hu et al. 2021). However, these approaches still require loading and serving the full base model at inference, so memory footprint, cold-start time, and per-task model variants remain the dominant cost drivers - especially when many tasks must be supported concurrently (a common reality for educational platforms). We therefore pose the central question: for challenging cross-prompt trait-scoring scenarios, including those in AES/ASAG (Yadav and Team 2023; Katuka, Gain, and Yu 2024), *can a single and compact model support multiple scoring tasks without material loss in performance?*

To answer this question, we propose UniMoE: a knowledge-distilled multi-task method based on a Mixture-of-Experts (MoE) architecture that trains a single compact model for multiple tasks. A multi-task (Zhang and Yang 2018; Fang et al. 2025a) backbone learns representations common to automated scoring, while lightweight task heads produce rubric-specific predictions (e.g., holistic and trait scores). Between them, a gated MoE module (Masoudnia and Ebrahimpour 2014; Fedus, Dean, and Zoph 2022) routes each example to experts with different weights, preserving item/trait idiosyncrasies and mitigating negative transfer across heterogeneous tasks. Practically, UniMoE consolidates storage and serving into a single backbone, minimizes per-task additions, and adapts to new tasks more reliably than standard multi-task models. By providing specialized capacity for rubric-specific patterns, the MoE layer reduces

*This is the corresponding author.

inter-task interference and enhances transfer learning: new tasks reuse shared representations while leveraging specialized experts, requiring only small task-specific components without destabilizing prior tasks. The result is faster, lower-cost adaptation at scale without sacrificing rubric fidelity.

In settings where high-capacity per-task reference models ('teachers') exist or can be trained offline, but deployment requires a single lightweight model, we extend UNIMOE to UNIMOE-GUIDED via knowledge distillation (Gou et al. 2021; Fang et al. 2025b). The compact multi-task model used at inference serves as the 'student'. For each task, the teacher provides soft guidance (e.g., probability predictions), and the student minimizes a convex combination of the supervised loss and a distillation loss that aligns its outputs with the teacher's; the distillation term acts as a regularizer. This transfers knowledge from the teachers into the student, narrowing the gap to task-specialized models while preserving a single low-latency model with substantially lower storage and serving costs.

We evaluate UNIMOE-GUIDED on NGSS-aligned, multi-label science-reasoning tasks from the PASTA corpus, using seven primary tasks for model comparison and held-out tasks to probe generalization. Briefly, results show that UNIMOE-GUIDED achieves comparable results to training separate models, while being $\sim 6\times$ smaller than maintaining separate models and $87\times$ smaller than the teacher model. It also remains deployment-friendly, underscoring its practical deployability. Additionally, the generalization tasks show that UNIMOE-GUIDED achieves significantly better performance than simple MTL, emphasizing its superior generalization capability in handling unseen tasks.

The proposed UNIMOE-GUIDED method supports resource-efficient AI for educational automated scoring and other cost-sensitive domains, contributing to advancements in scalable, efficient model deployment. Our key contributions are summarized below:

- **Single deployable multi-task scorer.** A shared encoder plus a task-aware MoE and lightweight heads balance shared structure with rubric-specific specialization, mitigating negative transfer across diverse assessments.
- **Teacher-guided compression.** Distillation from task-specialized teachers into one compact student yields the highest average reliability on our seven-task benchmark, while reducing storage by $\sim 6\times$ versus maintaining separate per-task students and by $87\times$ versus the 20B teacher.
- **Robust extension to new tasks.** Adding a new head and lightly tuning a small portion of the MoE integrates new assessments without retraining the backbone, outperforming plain MTL and narrowing the gap to per-task models on held-out tasks.

Related Work

Multi-Task Learning (MTL). MTL has long been explored as a means to improve generalization by leveraging inductive biases shared across related tasks (Caruana 1997). Traditional approaches rely on hard or soft parameter sharing within a shared encoder, while more recent methods incorporate optimization-aware strategies, adaptive task weighting, and explicit modeling of inter-task relationships, particularly in the era of large foundation models (Yu et al. 2024). In the context of educational assessment, jointly training across rubric dimensions (multi-label) and prompts (multi-task) can increase efficiency but risks negative transfer when task distributions diverge. To mitigate this, modern MTL designs often combine a shared backbone with lightweight task-specific modules or routing mechanisms to preserve both efficiency and task fidelity (Jacob et al. 2023; Auty et al. 2024). This motivates our choice of a shared encoder augmented with task-aware specialization capacity.

Mixture-of-Experts (MoE) Architectures. MoE models address the limitations of fixed shared capacity in MTL by increasing total model capacity while keeping per-example computation low through conditional routing of tokens to a small subset of experts (Shazeer et al. 2017). Large-scale deployments such as GShard and Switch Transformers demonstrated the scalability of gating paired with auxiliary load-balancing losses (Fedus, Zoph, and Shazeer 2022). Subsequent advances, including expert-choice routing (Zhou et al. 2022) and state-of-the-art MoE implementations in Mixtral, DeepSeek-V2/V3, and Qwen-2.x (Jiang et al. 2024; Liu et al. 2024a,b; Yang et al. 2024, 2025), highlight the potential of weighted activation to enable diverse specializations without prohibitive costs.

Knowledge Distillation (KD). Even with efficient architectures like MoE, deploying large models in educational settings often requires further compression. KD offers a principled solution by transferring the behavior of high-capacity teacher models into smaller, faster student models (Gou et al. 2021; Fang et al. 2025b, 2024). The field has evolved from early logit-matching to feature-based transfer, self-distillation, and task-aware distillation strategies. For multi-label scenarios such as rubric scoring, specialized KD methods mitigate label competition by distilling one-vs-rest probabilities or label-wise embeddings (Yang et al. 2023), while in multi-task contexts, cross-task KD improves stability and transferability under distribution shifts (Jacob et al. 2023; Auty et al. 2024). In educational assessment, recent studies have applied KD to fine-tune LLMs for scoring under latency and fairness constraints (Latif et al. 2024b; Misgna et al. 2024). Building on this foundation, our approach distills knowledge from multiple strong task-specific teachers into a single multi-task MoE student.

Methodology

In this section, we describe the proposed UNIMOE-GUIDED method, which constructs a single student model for automated scoring across multiple educational tasks. The resulting model consists of three components: a shared encoder that captures general linguistic representations, a task-aware Mixture-of-Experts (MoE) layer that activates a small set of specialized experts for each response, and task-specific heads that generate rubric-aligned predictions. To train the student, UNIMOE-GUIDED combines authentic labels with KD from high-capacity teacher models, allowing the student to approach teacher-level accuracy while remaining compact and practical for classroom deployment.

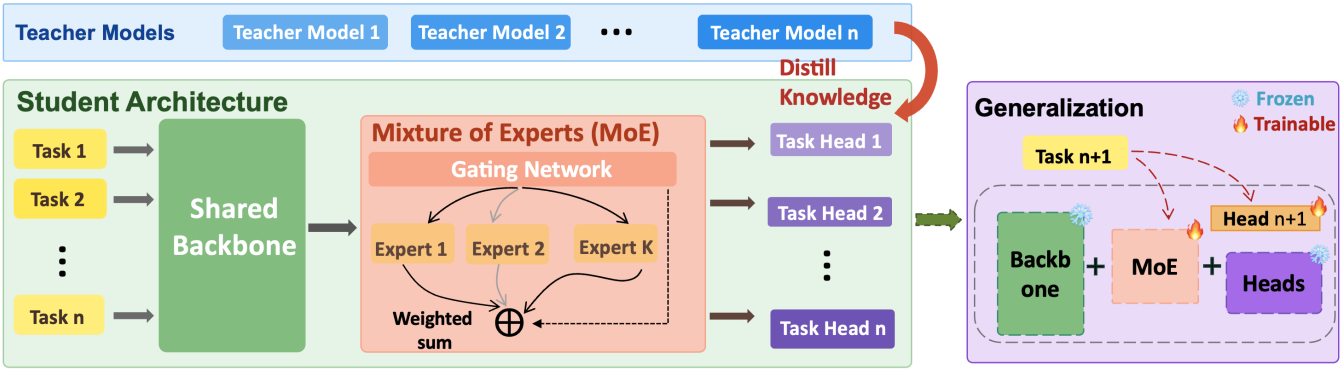

Figure 1: Workflow of UNIMOE and its teacher-guided variant UNIMOE-GUIDED. Left and center: multiple tasks share a single backbone; a MoE module with a learned gating network routes representations to a small subset of experts and aggregates them by a weighted sum; the outputs feed lightweight task heads. In UNIMOE-GUIDED, teacher models provide soft targets during training. Right: for a new task, the backbone and existing heads stay frozen, and only a new head and optionally a small portion of the MoE are trained, enabling rapid, low-overhead adaptation.

MoE-Based Multi-Task Learning with Knowledge Distillation

We model automated scoring of middle-school science explanations as multi-label prediction across T NGSS-aligned tasks. For a student response X_t from task t with binary rubric indicators $y_t \in \{0,1\}^{D_t}$ the model outputs logits $z_t(X_t) \in \mathbb{R}^{D_t}$ and per-indicator probabilities

$$p_t(X_t) = \sigma(z_t(X_t)),$$

where $\sigma(\cdot)$ denotes the element-wise sigmoid function.

Architecture. The proposed architecture, UNIMOE-GUIDED, includes (i) a shared Transformer encoder, (ii) a MoE block with task-aware routing, and (iii) compact task-specific heads. The shared encoder $f_\theta(\cdot)$ shares parameters across all tasks to provide a common linguistic representation, enabling a single deployable checkpoint. The MoE block combines multiple experts via a gating network that produces varying weights per input, allowing experts to specialize in rubric-specific language and reasoning patterns. Lastly, the task-specific heads are tailored to each task, maintaining rubric fidelity and strong per-task accuracy. The heads remain small, so adding a new rubric increases storage only by a lightweight head rather than a full model. We introduce the details of the three parts below.

Shared encoder. We use a pre-trained BERT-base model (Devlin et al. 2019) as a shared text encoder, fine-tuned on our tasks, that maps a student response X to contextual representations $H_t = f_\theta(X_t)$, where $H_t \in \mathbb{R}^{L\times d}$ with sequence length L, and hidden size d. This task-agnostic encoder captures linguistic patterns common across rubrics and is reused for every task, reducing storage and simplifying deployment.

MoE with task-aware routing. We append a MoE block after the encoder to enable dynamic routing of representations to specialized experts, improving task-specific adaptation. The MoE block comprises M experts $\{E_j\}_{j=1}^{M}$, where each expert is a two-layer feed-forward network with nonlinearity and dropout, applied token-wise to the contextual representation H_t, resulting $E_j(H_t) \in \mathbb{R}^{L\times d}$.

A task-aware gating network produces mixture weights over experts. We form the gate input by concatenating a pooled sequence summary $h_{\text{t,pool}} = \pi(H_t) \in \mathbb{R}^d$ with a learned task embedding e_t, yielding $m_t = [h_{\text{t,pool}}\,; e_t]$. The gating network $g(\cdot)$ maps m_t to logits $G_t = (g_{t,1},\ldots,g_{t,M})^\top \in \mathbb{R}^M$, which are converted to weights $\alpha_t = (\alpha_{t,1},\ldots,\alpha_{t,M})^\top$ by a softmax,

$$\alpha_{t,j} = \frac{\exp(g_{t,j})}{\sum_{j=1}^{M}\exp(g_{t,j})}.$$

These weights are shared across tokens, yielding MoE-enhanced representation $\tilde{H}_t$:

$$\tilde{H}_t = \sum_{j=1}^{M} \alpha_{t,j}\, E_j(H_t).$$

To discourage expert collapse and promote coverage, we add a load-balancing penalty that encourages the average gate weights to match the uniform distribution. For the responses of size B, let $\alpha_j^{(b)}$ be the weight assigned to expert j for example b, and define $\bar{\alpha}_j = \frac{1}{B}\sum_{b=1}^{B}\alpha_j^{(b)}$. The load-balancing penalty is

$$\mathcal{L}_{\text{LB}} = \frac{1}{M}\sum_{j=1}^{M}\left(\bar{\alpha}_j - \frac{1}{M}\right)^2.$$

This regularizer encourages broader expert utilization. The mechanisms of expert structure, task-aware gating, and explicit balancing are essential for stable specialization across heterogeneous rubrics.

Task-specific heads. For each task t with D_t rubric labels, a small MLP head $h_t(\cdot)$ maps the MoE-enhanced representation $\tilde{H}$ to logits $z_t \in \mathbb{R}^{D_t}$. Only heads are task-specific, so adding a rubric requires instantiating a new head (and, if desired, modest MoE tuning) rather than a new model.

Training objectives. The NGSS-aligned assessments are multi-label: each label is binary, and multiple labels may

be present simultaneously. We therefore optimize a binary cross-entropy (BCE) objective:

$$\mathcal{L}_{\text{task}} = E_{(t,X_t)}(\sum_{c=1}^{D_t} \big[- \, y_{t,c} \, \log \sigma(z_{t,c}) \\ - (1 - y_{t,c}) \, \log(1 - \sigma(z_{t,c}))\big]),$$

where $y_t \in \{0,1\}^{D_t}$ are the ground-truth rubric indicators for task t. In practice, we implement the expectation as an average over tasks and samples. This loss directly models the presence or absence of multiple skills or evidence types in a single response.

Knowledge distillation. To reduce storage while preserving agreement with expert scoring, we integrate KD from pre-trained teacher models. Given teacher logits $z^{(\text{teacher})}$, which are precomputed and loaded during training, and a temperature $\tau > 0$, we define softened targets $q^\tau = \sigma\left(z^{(\text{teacher})}/\tau\right)$ and softened student outputs $p^{(\tau)} = \sigma\left(z_t/\tau\right)$, and minimize a distillation BCE:

$$\mathcal{L}_{\text{KD}} = E_{(t,X_t)}\left(\text{BCE}\left(p^{(\tau)}, q^\tau\right) \cdot \tau^2\right).$$

KD transfers rubric-specific decision boundaries to the student, stabilizes learning when some labels are sparse, and enables efficient deployment without reintroducing teachers at inference.

Overall training objective. We minimize the average per-example objective across all tasks and samples, plus the load-balancing penalty, and the distillation regularization:

$$\mathcal{L} = \mathcal{L}_{\text{task}} + \lambda_{\text{LB}}\mathcal{L}_{\text{LB}} + \lambda_{\text{KD}}\mathcal{L}_{\text{KD}},$$

where λ_{LB} and λ_{KD} are weights controlling the contribution of load-balancing penalty and distillation regularization.

Generalization

For a new assessment task, we extend the trained multi-task MoE model in two steps that preserve prior competence while enabling fast adaptation. We append a lightweight, task-specific prediction head for the new task. During adaptation, the shared backbone remains frozen to preserve linguistic representations; we update only the MoE layers (experts and router) and the newly added head. Training uses labeled data from the new task together with a small rehearsal subset from previously seen tasks to mitigate forgetting. When teacher signals are available, we reuse the same distillation objective defined in base training; otherwise, we optimize the supervised task loss with the existing load-balancing regularizer to prevent expert collapse. Only a small fraction of parameters is updated, enabling rapid, stable inclusion of new assessments while maintaining performance on earlier tasks.

In summary, UNIMOE-GUIDED combines a shared encoder that provides transferable linguistic representations, a task-aware, load-balanced MoE that delivers scalable specialization, and compact task heads that capture rubric differences with minimal storage. KD then consolidates teacher knowledge into a single deployable student, well-suited to resource- and privacy-constrained classroom settings.

Dataset	No. Labels	Training size	Testing size
Task 1	4	955	239
Task 2	4	666	167
Task 3	3	958	240
Task 4	10	956	240
Task 5	5	836	210
Task 6	6	653	164
Task 7	5	956	239
Task 8	5	956	240
Task 9	3	1111	278

Table 1: Statistics of the nine NGSS-aligned assessment tasks from the PASTA corpus.

Dataset Details

We use pre-existing responses from U.S. middle-school classrooms (grades 6–8) to nine NGSS-aligned assessment tasks developed under the NGSA initiative (Harris, Krajcik, and Pellegrino 2024; PASTA November, 2023). Each response is scored across multiple rubric dimensions, with each label treated as binary, indicating whether a given dimension is satisfied. The tasks were designed to elicit evidence of three-dimensional learning as defined by the NGSS, integrating Disciplinary Core Ideas (DCIs), Crosscutting Concepts (CCCs), and Science and Engineering Practices (SEPs). In particular, they align with MS-PS1-2, where students analyze and interpret data on the properties of substances before and after interaction to determine whether a chemical reaction has occurred (Council et al. 2013).

For each task, approximately 1,200 students' responses were randomly selected from a large pool of responses, and after data cleaning, each task included slightly fewer responses; exact per-task counts and train/test splits are reported in Table 1 (Zhai, He, and Krajcik 2022). All responses were anonymized, and no demographic variables were included. The sample reflects geographic diversity across schools to be statistically representative of the broader U.S. population.

The tasks draw on fundamental concepts of chemistry within the physical sciences domain, specifically "Chemical Reactions". They require students to analyze data and distinguish substances by properties, capturing multi-dimensional reasoning in real-world contexts. Automated rubric-based scoring provides diagnostic information, highlighting areas where students may need additional support and offering educators actionable insights into student understanding.

Task Example. For instance, in Task 3, students were asked to identify gases in an experiment by comparing their properties to those listed in a data table (see Fig. 2). Successfully completing this task required understanding the structure and properties of matter, knowledge of chemical reactions, and the ability to plan investigations while recognizing patterns in the data.

A structured scoring rubric was developed to encompass five response dimensions aligned with the science learning framework: SEP+DCI, SEP+CCC-1, SEP+CCC-2, DCI-

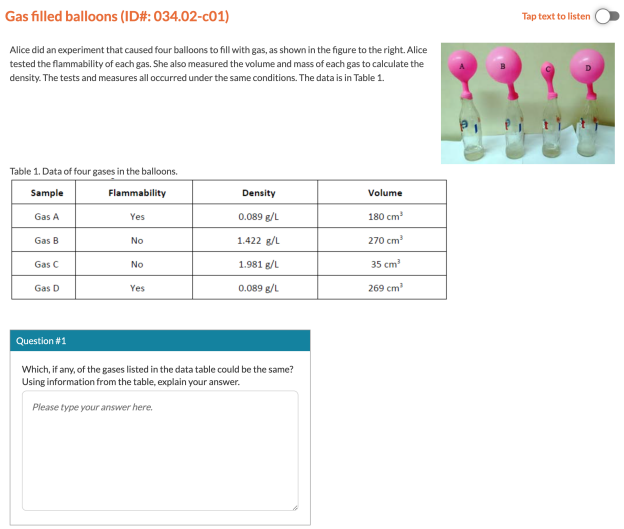

Figure 2: Illustrative Multi-label Task: Gas-Filled Balloons

ID	Perspective	Description
E1	SEP+DCI	Student states that Gas A and D could be the same substance.
E2	SEP+CCC-1	Student describes the pattern (comparing data in different columns) in the flammability data of Gas A and Gas D as the same.
E3	SEP+CCC-1	Student describes the pattern (comparing data in different columns) in the density data of Gas A and Gas D, which is the same in the table.
E4	DCI-2	Student indicates flammability is one characteristic of identifying substances.
E5	DCI-2	Student indicates that density is one characteristic of identifying substances.

Table 2: Scoring rubric for Task 3, Gas-filled balloons.

1, and DCI-2. The rubric was designed to capture multi-dimensional cognitive processes (He et al. 2024). Table 2 outlines the specific criteria for each dimension. Students were assessed simultaneously across these perspectives, receiving scores that reflected their understanding of DCIs, CCCs, and SEPs as defined by the rubric. To enhance the validity of these multi-perspective rubrics, the research team collaborated closely with experienced science educators.

Experiments

We evaluate on seven NGSS-aligned, multi-label scoring tasks (Tasks 1–7), and reserve two additional tasks (Tasks 8–9) to test adaptation to new assessments (see Table 1 for label counts and split sizes). Each task consists of short, open-ended science responses with rubric-aligned binary indicators (multi-label). We use the official train/test partitions and carve a stratified validation split (10%) from training for model selection. Unless noted, all experiments use BERT-base-uncased as the backbone encoder.

Baselines

- **Individual (Per-Task).** Separate BERT-base models fine-tuned per task (upper-bound capacity).
- **MTL (Shared Encoder + Heads).** A single BERT-base shared encoder with lightweight task-specific classification heads; no MoE.
- **UniMoE (Ours, no KD).** The MTL backbone is augmented with a gated Mixture-of-Experts (MoE) block placed between the encoder and heads.
- **UniMoE-Guided (Ours).** UniMoE trained with KD from task-specialized teachers.

Evaluation Metrics

- **Educational Reliability.** We follow educational scoring practice and report: Cohen's kappa score (McHugh 2012) averaged over labels, Macro-F1, Micro-F1, and per-label accuracy averaged over labels. We use a fixed decision threshold of 0.5 across all labels.
- **Resource Efficiency.** We report efficiency (parameter count, disk size) and latency (per-sample inference time).
- **Generalization Ability.** We evaluate the model's performance on held-out tasks (Tasks 8–9) using the educational reliability criterion introduced earlier.

Model Structure. Our architecture builds on BERT-base-uncased as a shared encoder (109M, 91%). We employ two MoE modules (9M, 7.3%), with each containing a configurable number of experts (3–5 tested), with each expert designed as a two-layer feed-forward network. A gating mechanism with task-aware routing assigns inputs to experts, and load balancing is encouraged through a regularization penalty. Each task then uses a lightweight head, consisting of a linear projection, ReLU activation, dropout, and a final linear layer producing task-specific logits (2M, 1.7% for all heads). KD integrates pre-computed teacher probabilities using temperature-scaled binary cross-entropy.

Model Training. All methods share the same basic training setup. We use AdamW (lr $= 2 \times 10^{-5}$, batch size=32, max epochs=20) with linear learning rate warmup and decay, gradient clipping, and early stopping (patience=3). Input text is tokenized with the standard BERT tokenizer, truncated or padded to a maximum length of 100 tokens. Labels are represented as multi-hot binary vectors, and teacher probabilities are normalized for KD. Beyond this common setup, UniMoE and UniMoE-Guided require additional hyperparameter tuning because they integrate knowledge from multiple datasets and rely on weighting mechanisms. For UniMoE, we search over the number of experts M ($\{3, 4, 5\}$) and load balance weight λ_{LB} ($\{0.005, 0.01, 0.05\}$). For KD-UniMoE, we additionally tune the KD weight λ_{KD} ($\{0.05, 0.1, 0.3, 0.5\}$). Experiments are run on NVIDIA Tesla V100 GPUs.

Teacher Models (20B). For each task, we fine-tune the 20B-parameter `openai/gpt-oss-20b` model using parameter-efficient LoRA adapters (rank $r = 32$, $\alpha = 64$), which update about 1% of the model's weights while keeping the rest frozen. Training is performed on an NVIDIA

Metric	Method	Task 1	Task 2	Task 3	Task 4	Task 5	Task 6	Task 7	Average
Cohen's κ ↑	Individual	0.5164	0.5526	0.8638	0.7140	0.6863	0.6480	0.5740	0.6510
	MTL	0.4277	0.5052	0.8891	0.5238	0.5782	0.4393	0.5281	0.5559
	UniMoE (no KD)	0.4457	0.5454	**0.9091**	0.6418	0.7340	0.6012	**0.6063**	0.6405
	UniMoE-Guided	**0.5246**	**0.5691**	0.8830	**0.6499**	**0.7545**	**0.6288**	0.5787	**0.6555**
Macro-F1 ↑	Individual	0.5636	0.6725	0.9574	0.7952	0.7439	0.8007	0.6831	0.7444
	MTL	0.4832	0.6325	0.9662	0.6909	0.6441	0.6644	0.6415	0.6747
	UniMoE (no KD)	0.4857	0.6659	**0.9727**	0.7541	0.7952	0.7697	**0.7027**	0.7351
	UniMoE-Guided	**0.5806**	**0.6958**	0.9657	**0.7603**	**0.8133**	**0.7939**	0.6859	**0.7565**
Micro-F1 ↑	Individual	0.8859	0.7725	0.9577	0.8746	0.7891	0.8551	0.7469	0.8290
	MTL	0.8740	0.7860	0.9660	0.7813	0.7236	0.8096	0.7341	0.8107
	UniMoE (no KD)	**0.8911**	0.7443	0.9620	**0.8360**	0.8089	0.8422	**0.7669**	0.8359
	UniMoE-Guided	0.8727	**0.7884**	**0.9649**	0.8349	**0.8214**	**0.8487**	0.7490	**0.8400**
Per-Label Accuracy ↑	Individual	0.9383	0.8563	0.9431	0.9100	0.9038	0.8750	0.8452	0.8923
	MTL	0.9310	0.8533	0.9542	0.8379	0.8705	0.8303	0.8351	0.8732
	UniMoE (no KD)	**0.9425**	0.8488	0.9525	**0.8846**	0.9095	0.8648	**0.8611**	**0.8948**
	UniMoE-Guided	0.9289	**0.8578**	**0.9528**	0.8825	**0.9143**	**0.8699**	0.8469	0.8933

Table 3: Results on Tasks 1–7 across four methods. Each column reports performance per task, with averages shown in the final column. Metrics include Cohen's κ, Macro-F1, Micro-F1, and Per-Label Accuracy. Best results per task are bolded.

H100 GPU in mixed precision. The resulting teachers provide task-specific probability distributions, which are then distilled into UniMoE-Guided training.

Results

To provide insight that directly calling commercial LLM APIs is insufficient for our educational setting, we evaluate few-shot prompting of the 20B teacher model using the question, analytic rubric, and a concise exemplar response. As detailed in the Supplementary Materials, the resulting average Cohen's κ is only about 0.05, demonstrating that off-the-shelf LLM prompting performs poorly on our tasks and that fine-tuned, rubric-aware models remain essential.

Educational reliability. Table 3 summarizes the results across all evaluation metrics and tasks. Several key trends emerge. First, the naive UniMoE consistently outperforms the standard MTL baseline. For example, Cohen's κ improves from 0.556 to 0.641, indicating that expert routing effectively mitigates negative transfer and enhances rubric fidelity. Similar gains are observed for Macro-F1 (0.675 vs. 0.735) and Micro-F1 (0.811 vs. 0.836), confirming that MoE provides additional capacity without inflating computational cost. Second, incorporating knowledge distillation (KD) further improves model performance. The UniMoE-Guided variant achieves the highest average scores across κ, Macro-F1, and Micro-F1 metrics, while matching the naive UniMoE in per-label accuracy. Notably, it even surpasses individually fine-tuned per-task BERT models in some settings. For example, UniMoE-Guided attains an average κ of 0.656, exceeding the individual models' 0.651, and achieves the best Macro-F1 of 0.757 compared to 0.744 from individual models. These results demonstrate that distilling knowledge from large task-specific teachers into a single multi-task MoE student not only closes the performance gap but can also surpass individually trained models while offering superior efficiency in storage and deployment.

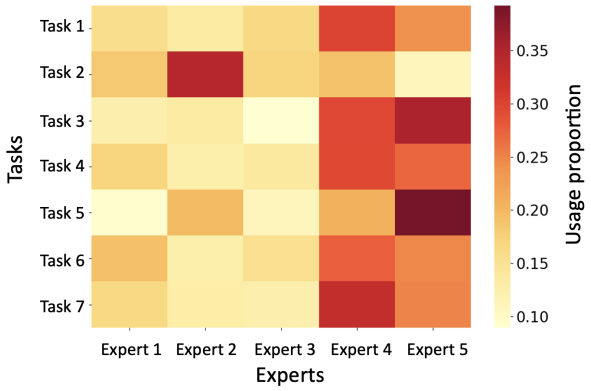

Figure 3: Expert usage patterns. Heatmap showing the proportion of each expert's usage for each task.

Figure 3 shows the expert usage distribution across tasks in our UniMoE-Guided model. The results indicate that different tasks rely on experts with distinct preferences rather than uniformly distributing attention. For example, Task 2 heavily utilizes Expert 2 (34.5%), while Task 5 shows a strong reliance on Expert 5 (39.2%). Other tasks, such as Task 1 and Task 6, exhibit a more balanced distribution across multiple experts. This demonstrates that the MoE structure successfully enables task-specific specialization while still allowing shared experts to contribute across tasks, thereby capturing both commonalities and differences among diverse scoring rubrics.

Resources Efficiency. Table 4 reports the efficiency and inference time of different model architectures. Our proposed UniMoE-Guided method achieves a compact size of 459 MB with 120M parameters, while maintaining an inference latency of 14 ms per sample. Compared to training individual BERT-based models for each task (772M parame-

Model	Parameters	Disk Size (MB)	Relative Size	Inference Time
UniMoE-Guided	120M	459	1.00x	14 ms/sample
Teacher (oss)	20B	40k (16-bit)	87.15x	43 ms/sample
Individual	772M	2945	6.42x	12 ms/sample
MTL	113M	426	0.93x	13 ms/sample

Table 4: Model size and inference latency. Relative size is normalized to UNIMOE-GUIDED. Latency is measured on a V100 GPU for all student models and on an H100 GPU for the teacher (not directly comparable).

ters, 2945 MB), UNIMOE-GUIDED reduces disk storage by more than 6×. Similarly, relative to a simple MTL model (113M parameters, 426 MB), UNIMOE-GUIDED shows nearly identical latency (14 ms vs. 13 ms), demonstrating that the additional MoE structure does not incur significant runtime cost. In contrast, the large GPT-based teacher model (20B parameters, 40 GB at FP16 on an H100 GPU) requires 87× more storage, highlighting the impracticality of direct deployment in the classroom or large-scale educational settings. While the inference time is reported, we load the model on an H100 due to its size, whereas student models run on a V100; therefore, the latency numbers are not directly comparable. These results confirm that UNIMOE-GUIDED strikes a favorable balance between efficiency and inference speed, while retaining the benefits of knowledge distilled from powerful teachers.

Generalization

We evaluate model generalization on two new tasks (Task 8 and Task 9), comparing against traditional MTL and individually trained models (gold standard). In MTL, a new head is added with the backbone frozen. In UNIMOE-GUIDED, the backbone remains frozen while both a new head and MoE layers are fine-tuned, yielding ∼7.5% trainable parameters. Results, as shown in Table 5, demonstrate that UNIMOE-GUIDED substantially outperforms MTL across all metrics and approaches the performance of individual models. For example, in Task 9, Cohen's κ improves from 0.2221 (MTL) to 0.4540, and Macro F1 from 0.2707 to 0.5592, more than doubling generalization quality. Fine-tuning the MoE layers induces slight decreases on old tasks (Cohen's κ: 0.6555 → 0.6279; Macro-F1: 0.7565 → 0.7322; Micro-F1: 0.8400 → 0.8258; Per-label acc: 0.8933 → 0.8870), yet performance remains consistently higher than MTL. Overall, these results underscore the method's efficiency and robustness, showing that UNIMOE-GUIDED achieves strong generalization without suffering from the severe degradation typical of conventional MTL.

Metric	Method	Task 8	Task 9
Cohen's κ ↑	Individual	0.6724	0.6314
	MTL	0.2954	0.2221
	UniMoE-Guided	0.4680	0.4540
Macro-F1 ↑	Individual	0.7696	0.7130
	MTL	0.5868	0.2707
	UniMoE-Guided	0.6513	0.5592
Micro-F1 ↑	Individual	0.8595	0.7196
	MTL	0.7077	0.2736
	UniMoE-Guided	0.7543	0.5697
Per-Label Accuracy ↑	Individual	0.8858	0.8729
	MTL	0.7625	0.8153
	UniMoE-Guided	0.8100	0.8297

Table 5: Generalization performance on held-out tasks (Task 8-9). Per-task models serve as an upper bound; MTL struggles to adapt, while UNIMOE-GUIDED approaches per-task performance with only 7.5% of parameters updated.

Conclusion

We presented UNIMOE-GUIDED, a knowledge-distilled multi-task Mixture-of-Experts scorer designed for resource-constrained educational deployment. By pairing a shared encoder with weighted experts and small task heads - and distilling from task-specific teachers - the approach matches or exceeds per-task baselines at a fraction of the cost. Evaluated across seven NGSS-aligned science tasks, UNIMOE-GUIDED achieves comparable or superior performance to individually fine-tuned models (e.g., κ: 0.656 vs. 0.651; Macro-F1: 0.757 vs. 0.744), despite being ∼6× smaller than separate task-specific models and ∼87× smaller than the original 20B-parameter teacher model. This compact single-checkpoint design ensures practical classroom deployment, balancing speed, storage efficiency, and scoring accuracy. A key advantage of UNIMOE-GUIDED is its extensibility. For new assessments, simply adding a task head and lightly adapting the MoE (with the backbone frozen) yields significant improvements over MTL baselines—narrowing the gap to per-task specialization (e.g., held-out task performance: Cohen's κ 0.454 vs. 0.222 for MTL; Macro-F1 0.559 vs. 0.271). While minor regressions occur on prior tasks during adaptation, performance remains consistently above MTL levels, enabling low-effort incorporation of new rubrics without compromising existing capabilities.

Limitations and future work. We aim to (i) strengthen lifelong learning to further curb residual forgetting during task additions, (ii) improve interpretability of expert specialization and explore rubric-aware routing, and (iii) enhance the KD strategy to better distill knowledge from teachers. Together, these directions aim to strengthen the reliability and reach of automated scoring systems that must scale across grades, standards, and classrooms while remaining efficient and deployable.

Acknowledgments

This work was partially supported by the Institute of Education Sciences (IES) [R305C240010], the U.S. National Science Foundation (NSF) [DMS-2138854, DMS-1925066, DMS-1903226, DMS-2124493, DMS-2311297, DMS-2319279, DMS-2318809, 2101104], and the National Institutes of Health (NIH) [R01GM152814]. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the IES, NSF, or NIH.

References

Auty, D.; et al. 2024. Learning to Project for Cross-Task Knowledge Distillation. In *BMVC*.

Caruana, R. 1997. Multitask Learning. *Machine Learning*, 28(1): 41–75.

Council, N. R.; et al. 2013. Next generation science standards: For states, by states.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 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)*, 4171–4186.

Dikli, S. 2006. An overview of automated scoring of essays. *The Journal of Technology, Learning and Assessment*, 5(1).

Fang, L.; Chen, Y.; Zhong, W.; and Ma, P. 2024. Bayesian knowledge distillation: A bayesian perspective of distillation with uncertainty quantification. In *Forty-first International Conference on Machine Learning*.

Fang, L.; Latif, E.; Lu, H.; Zhou, Y.; Ma, P.; and Zhai, X. 2025a. Efficient multi-task inferencing: Model merging with gromov-wasserstein feature alignment. In *International Conference on Artificial Intelligence in Education*, 192–200. Springer.

Fang, L.; Yu, X.; Cai, J.; Chen, Y.; Wu, S.; Liu, Z.; Yang, Z.; Lu, H.; Gong, X.; Liu, Y.; et al. 2025b. Knowledge distillation and dataset distillation of large language models: Emerging trends, challenges, and future directions. *arXiv preprint arXiv:2504.14772*.

Fedus, W.; Dean, J.; and Zoph, B. 2022. A review of sparse expert models in deep learning. *arXiv preprint arXiv:2209.01667*.

Fedus, W.; Zoph, B.; and Shazeer, N. 2022. Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity. *Journal of Machine Learning Research*, 23(120): 1–39.

Gou, J.; Yu, B.; Maybank, S. J.; and Tao, D. 2021. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6): 1789–1819.

Han, Z.; Gao, C.; Liu, J.; Zhang, J.; and Zhang, S. Q. 2024. Parameter-efficient fine-tuning for large models: A comprehensive survey. *arXiv preprint arXiv:2403.14608*.

Harris, C. J.; Krajcik, J. S.; and Pellegrino, J. W. 2024. *Creating and using instructionally supportive assessments in NGSS classrooms*. NSTA Press.

He, P.; Shin, N.; Zhai, X.; and Krajcik, J. 2024. Guiding Teacher Use of Artificial Intelligence-Based Knowledge-in-Use Assessment to Improve Instructional Decisions: A Conceptual Framework. In Zhai, X.; and Krajcik, J., eds., *Uses of Artificial Intelligence in STEM Education*, xx–xx. Oxford University Press.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Jacob, G. M.; et al. 2023. Online Knowledge Distillation for Multi-Task Learning. In *WACV*.

Jescovitch, L. N.; Scott, E. E.; Cerchiara, J. A.; Merrill, J.; Urban-Lurain, M.; Doherty, J. H.; and Haudek, K. C. 2021. Comparison of machine learning performance using analytic and holistic coding approaches across constructed response assessments aligned to a science learning progression. *Journal of Science Education and Technology*, 30(2): 150–167.

Jiang, A. Q.; Sablayrolles, A.; Roux, A.; Mensch, A.; Savary, B.; Bamford, C.; Chaplot, D. S.; de las Casas, D.; Bou Hanna, E.; Bressand, F.; et al. 2024. Mixtral of Experts. *arXiv preprint arXiv:2401.04088*.

Katuka, G. A.; Gain, A.; and Yu, Y.-Y. 2024. Investigating Automatic Scoring and Feedback using Large Language Models. *arXiv preprint arXiv:2405.00602*.

Latif, E.; Fang, L.; Ma, P.; and Zhai, X. 2024a. Knowledge distillation of llms for automatic scoring of science assessments. In *International Conference on Artificial Intelligence in Education*, 166–174. Springer.

Latif, E.; et al. 2024b. Fine-tuning ChatGPT for Automatic Scoring. *Computer Assisted Language Learning*. Available via Elsevier/ScienceDirect.

Lee, G.-G.; Latif, E.; Wu, X.; Liu, N.; and Zhai, X. 2024. Applying large language models and chain-of-thought for automatic scoring. *Computers and Education: Artificial Intelligence*, 6: 100213.

Liu, A.; et al. 2024a. DeepSeek-V2: A Strong, Economical, and Efficient Mixture-of-Experts Language Model. *arXiv preprint arXiv:2405.04434*.

Liu, A.; et al. 2024b. DeepSeek-V3 Technical Report. *arXiv preprint arXiv:2412.19437*.

Liu, X.; Sun, T.; He, J.; Wu, J.; Wu, L.; Zhang, X.; Jiang, H.; Cao, Z.; Huang, X.; and Qiu, X. 2021. Towards efficient NLP: A standard evaluation and a strong baseline. *arXiv preprint arXiv:2110.07038*.

Mahmoud, S.; Nabil, E.; and Torki, M. 2024. Automatic Scoring of Arabic Essays: A Parameter-Efficient Approach for Grammatical Assessment. *IEEE Access*.

Masoudnia, S.; and Ebrahimpour, R. 2014. Mixture of experts: a literature survey. *Artificial Intelligence Review*, 42(2): 275–293.

McHugh, M. L. 2012. Interrater reliability: the kappa statistic. *Biochemia medica*, 22(3): 276–282.

Misgna, H.; et al. 2024. A Survey on Deep Learning-Based Automated Essay Scoring and Feedback Generation. *Artificial Intelligence Review*.

Moharreri, K.; Ha, M.; and Nehm, R. H. 2014. EvoGrader: an online formative assessment tool for automatically evaluating written evolutionary explanations. *Evolution: Education and Outreach*, 7(1): 15.

Page, E. B. 1966. The imminence of... grading essays by computer. *The Phi Delta Kappan*, 47(5): 238–243.

PASTA, P. T. November, 2023. Supporting Instructional Decision Making: Potential of An Automatically Scored Three-dimensional Assessment System. *https:/ai4stem.org/pasta/*.

Reidy, B. C.; Mohammadi, M.; Elbtity, M. E.; and Zand, R. 2023. Efficient deployment of transformer models on edge tpu accelerators: A real system evaluation. In *Architecture and System Support for Transformer Models (ASSYST@ ISCA 2023)*.

Shazeer, N.; Mirhoseini, A.; Maziarz, K.; Davis, A.; Le, Q.; Hinton, G.; and Dean, J. 2017. Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer. In *ICLR*.

Yadav, P.; and Team. 2023. TIES-MERGING: A Pruning-Based Approach to Alleviate Conflicts in Model Merging. In *Proceedings of the International Conference on Machine Learning (ICML)*, 4567–4578.

Yang, A.; Yang, B.; Hui, B.; Zheng, B.; Yu, B.; Zhou, C.; Li, C.; Li, C.; Liu, D.; Huang, F.; et al. 2024. Qwen2 Technical Report. *arXiv preprint arXiv:2407.10671*.

Yang, A.; et al. 2025. Qwen2.5 Technical Report. *arXiv preprint arXiv:2412.15115*.

Yang, P.; Xie, M.-K.; Zong, C.-C.; Feng, L.; Niu, G.; Sugiyama, M.; and Huang, S.-J. 2023. Multi-Label Knowledge Distillation. In *ICCV*.

Yu, J.; Dai, Y.; Liu, X.; Huang, J.; Shen, Y.; Zhang, K.; Zhou, R.; Adhikarla, E.; Ye, W.; Liu, Y.; et al. 2024. Unleashing the Power of Multi-Task Learning: A Comprehensive Survey Spanning Traditional, Deep, and Pretrained Foundation Model Eras. *arXiv preprint arXiv:2404.18961*.

Zhai, X.; He, P.; and Krajcik, J. 2022. Applying machine learning to automatically assess scientific models. *Journal of Research in Science Teaching*, 59(10): 1765–1794.

Zhang, Y.; and Yang, Q. 2018. An overview of multi-task learning. *National Science Review*, 5(1): 30–43.

Zhou, Y.; Lei, T.; Liu, H.; Du, N.; Huang, Y.; Zhao, V.; Dai, A.; Chen, Z.; Le, Q.; and Laudon, J. 2022. Mixture-of-Experts with Expert Choice Routing. In *NeurIPS*.