SRA: Span Representation Alignment for Large Language Model Distillation

Quoc Phong Dao[1]*, Hoang Son Nguyen[1]*, Pham Khanh Chi[1]*, Tung Nguyen[1], Linh Ngo Van[1,†], Diep Thi-Ngoc Nguyen[2], Trung Le[3]

[1]Hanoi University of Science and Technology,
[2]VNU University of Engineering and Technology, [3]Monash University

Abstract

Cross-Tokenizer Knowledge Distillation (CTKD) enables knowledge transfer between a large language model and a smaller student, even when they employ different tokenizers. While existing approaches mainly focus on token-level alignment strategies, which are often brittle and sensitive to discrepancies between tokenizers, we argue that the method of aggregating tokens into more robust representations before distillation is of equal importance. In this paper, we introduce **SRA** (**S**pan **R**epresentation **A**lignment for Large Language Model Distillation), a novel framework that reframes CTKD through the physical lens of Multi-Particle Dynamical Systems. SRA shifts the fundamental unit of alignment from tokens to robust, tokenizer-agnostic spans. We model each span as a cluster of particles and represent its state by its Center of Mass (CoM) - an attention-weighted average that captures rich semantic information. We leverage the concept of span centers of mass with attention-derived weighting to prioritize the most salient spans. In addition, we employ a geometric regularizer to preserve the structural integrity of the representation space and introduce aligned span logit distillation to enhance knowledge transfer across models. In challenging cross-architecture distillation experiments, SRA consistently and significantly outperforms state-of-the-art CTKD baselines, validating our physically-grounded approach.

1 Introduction

Large Language Models (LLMs) have achieved remarkable success, largely driven by scaling model parameters to billions or even trillions (DeepSeek-AI-Group, 2024; OpenAI, 2024). However, this immense scale poses significant challenges for practical deployment. Knowledge Distillation (KD), which transfers knowledge from a large teacher to a smaller student, has emerged as a critical technique for creating efficient yet powerful models (Hinton et al., 2015). While effective, conventional KD often assumes the teacher and student share an identical tokenizer, a restrictive assumption in practice (Sun et al., 2019; Sanh et al., 2020; Gu et al., 2024). Existing Cross-Tokenizer Knowledge Distillation (CTKD) methods tackle these mainly at the *token or logit level*: unifying output spaces via projections or cross-model attention (Zhang et al., 2024b), transporting probability mass between vocabularies with Optimal Transport (Boizard et al., 2025; Cui et al., 2025), or aligning token streams by edit distance (Wan et al., 2024; Chen et al., 2025).

We argue that a more robust solution requires a paradigm shift, grounding CTKD in the underlying dynamics of the Transformer architecture itself. The Transformer, with its layer-wise skip connections, can be interpreted as a discretized Ordinary Differential Equation (ODE), where each layer functions as a discrete step in time (Chen et al., 2018; Lu et al., 2019). In this ODE framework, the teacher's deep architecture provides a fine-grained discretization of the feature dynamics, while the student's shallower architecture is a coarser one. This perspective is powerfully complemented by the interpretation of Transformers as **Multi-Particle Dynamical Systems (MPDS)** (Lu et al., 2019), which models tokens as "particles" whose states (hidden representations) evolve through time (layers). Together, these views establish a clear principle: distillation should not be limited to final outputs, but must instead focus on transferring the *dynamics of the system's geometric evolution*. Recent work has begun to explore this, moving beyond static representations to distill the feature dynamics themselves (Gong et al., 2025). However, such methods are confined to same-tokenizer settings and do not address the fundamental geometric and granularity challenges of CTKD.

*Equal contribution
†Corresponding author: linhnv@soict.hust.edu.vn

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 32961–32975
July 2-7, 2026 ©2026 Association for Computational Linguistics

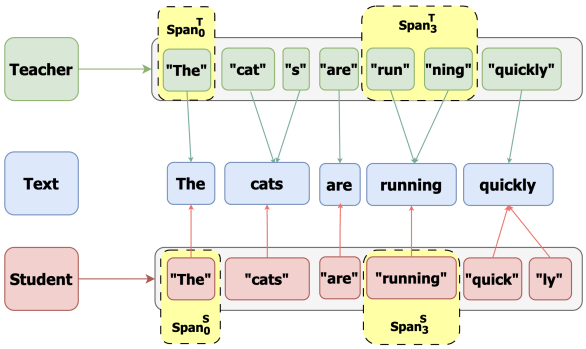

Figure 1: An illustration of the tokenizer mismatch between a teacher and a student model.

Grounded in this physical analogy, we introduce **SRA** (**S**pan **R**epresentation **A**lignment for Large Language Model Distillation), a novel framework that reframes CTKD. Instead of aligning brittle tokens, SRA aligns stable, tokenizer-agnostic spans (Figure 1). We conceptualize each span as a *cluster of particles* (its constituent tokens), whose state is represented by its *Center of Mass (CoM)* - an attention-weighted average of its particle positions. SRA distills these CoM representations and employs a geometric regularizer to preserve their relative positions in the representation space.

We demonstrate the effectiveness of SRA by transferring knowledge from powerful LLMs (Bai et al., 2023; Jiang et al., 2023; Yang et al., 2024) into smaller students (Radford et al., 2019; Zhang et al., 2024a, 2022) on multiple benchmarks. SRA consistently outperforms recent CTKD baselines (Wan et al., 2024; Zhang et al., 2024b; Cui et al., 2025; Boizard et al., 2025; Chen et al., 2025), and our ablations confirm that both the attention-weighted span representations and the geometric regularizer are critical to its success. In this work, we make the following contributions:

1. We introduce a novel conceptual framework for CTKD, grounded in the multi-particle view of Transformers (Lu et al., 2019). This allows us to model spans as particle clusters and leverage their **center of mass (CoM)** for stable, additive representations, moving beyond brittle token-level alignment and naturally supporting the transfer of dynamic, layer-to-layer feature evolution.

2. We propose to distill knowledge at the span level, encompassing both representations and logits. Specifically, we align span representations while preserving their geometric relationships. Furthermore, we introduce an attention-derived weighting scheme inspired by the Center of Mass (CoM) concept, which emphasizes the most salient spans and guides the student to focus on semantically important regions.

3. We evaluate SRA for cross-tokenizer and same tokenizer **decoder**→**decoder** distillation across diverse teacher–student pairs, demonstrating that SRA significantly outperforms recent CTKD baselines on multiple benchmarks. Extensive ablations validate the importance of our core components.

2 Related Work and Background

This section reviews prior work in knowledge distillation, focusing on techniques for same- and cross-tokenizer scenario.

2.1 Related Work

Knowledge Distillation and Cross-Tokenizer Methods Knowledge Distillation (KD) was pioneered by (Hinton et al., 2015) to transfer knowledge from a large teacher model to a smaller student by matching softened logit distributions. This was later extended to intermediate representations (Sun et al., 2019). These foundational methods and their modern generative counterparts, including highly successful models (Gu et al., 2024; Ko et al., 2024; Le et al., 2025), all operate under the crucial assumption that the teacher and student share an identical tokenizer. When this assumption is violated, the tokenizer mismatch presents a significant challenge. One line of research tackles this by seeking an explicit structural alignment between the disparate token sequences, using methods ranging from edit distance (Wan et al., 2024; Vu et al., 2026a) to more advanced techniques like entropy-weighted Dynamic Time Warping (Chen et al., 2025) or extending these ideas to preference alignment objectives (Nguyen et al., 2026). A second, more recent line of work bypasses discrete alignment altogether in favor of distributional or representational alignment. This includes projecting representations into a unified space (Zhang et al., 2024b) or, more powerfully, using Optimal Transport (OT) to match entire output distributions (Boizard et al., 2025; Cui et al., 2025; Vuong et al., 2026) and other advanced objectives like approximate likelihood matching (Minixhofer et al., 2025). Related representational alignment ideas have also

been explored for embedding model distillation (Truong et al., 2025; Vu et al., 2026b). While these methods have grown in sophistication, they often operate on the outputs of the tokenization process. They are therefore still fundamentally reliant on aligning sequences of discrete, sometimes brittle, token units or their distributions. In contrast, our SRA framework steps back from the token level entirely. It works at a more robust, semantically stable granularity: textual spans. We recover these spans from the raw text using character offsets, avoiding any need for direct token- or vocabulary-level alignment.

A deeper insight into the distillation can be gained by viewing the Transformer as a Multi-Particle Dynamical System (MPDS) (Lu et al., 2019). This framework establishes a compelling parallel between the structural components of a Transformer and the dynamics of interacting particles, as outlined in Table 7 (see Section 2.2.2).

2.2 Background

2.2.1 Knowledge Distillation Fundamentals

Knowledge Distillation (KD) (Hinton et al., 2015) is a well-established model compression paradigm in which a smaller student network is trained to approximate the behavior of a larger teacher. The learning objective typically augments the standard cross-entropy loss on ground-truth labels with a distillation term:

$$\mathcal{L} = \mathcal{L}_{CE}(y, z_s) + \lambda \mathcal{L}_{KD}(z_t, z_s), \qquad (1)$$

where z_t and z_s denote the teacher and student logits, and λ balances the distillation and supervised losses. The distillation term encourages the student's distribution to match a softened teacher output. For the distillation loss $\mathcal{L}_{KD}$, standard probabilistic measures such as KL divergence, Jensen–Shannon divergence, and cross-entropy are commonly employed.

2.2.2 Transformers as a Multi-Particle Dynamic System

A more profound understanding of the knowledge transfer process can be achieved by viewing the Transformer through the lens of a Multi-Particle Dynamical System (MPDS) (Lu et al., 2019). This framework draws a powerful analogy between the components of a Transformer and the physics of interacting particles, as summarized in Table 7 (Appendix 2.2.2). Grounding our design in the multi-particle dynamical system (MPDS) view of Transformers (Lu et al., 2019), where token hidden states are particle positions evolving by diffusion (attention) and convection (FFN), we argue: to distill dynamics, one must first preserve state geometry. Building on the MPDS view, Gong et al. (2025) pioneered the distillation of feature dynamics, aligning the *trajectory* and first-order derivatives of token representations by pushing them through each model's LM head and matching in the *vocabulary* (logit) space. While insightful, this work is limited to the same-tokenizer setting. To break the tokenizer barrier, our method treat each span as a cluster of particle, represented by their attention-weighted Center of Mass. We then learn the dynamics of the most salient spans.

3 Methodology

In this section, we present **Span Representation Alignment for Large Language Model Distillation (SRA)**, a framework for knowledge distillation. The core idea of SRA is to construct span-level representations for knowledge transfer, inspired by the center of mass in a multi-particle dynamic system (Section 3.1). We transfer span features from the teacher to the student (Sections 3.2 and 3.3), allowing the student to better capture the teacher's structural space in vector representations. The framework SRA is illustrated in Figure 2.

3.1 Span Representation Alignment

Although the teacher and student tokenizers often segment an input sentence into different token sequences, these tokens can still be grouped into spans that correspond to the same textual unit. Direct token-level alignment is therefore infeasible across tokenizers, but span-level alignment provides a natural bridge for transferring knowledge. We use such spans to serve as the basic units of alignment between the teacher and the student. The following subsections describe how spans are aligned across models and how span representations are formed.

Span Mapping via Longest Common Subsequence (LCS)

To construct and align spans, we start from an input sequence x that is tokenized by both the teacher and the student tokenizers. Each tokenizer produces a sequence of tokens together with their character offsets indicating the end position of each token in x,

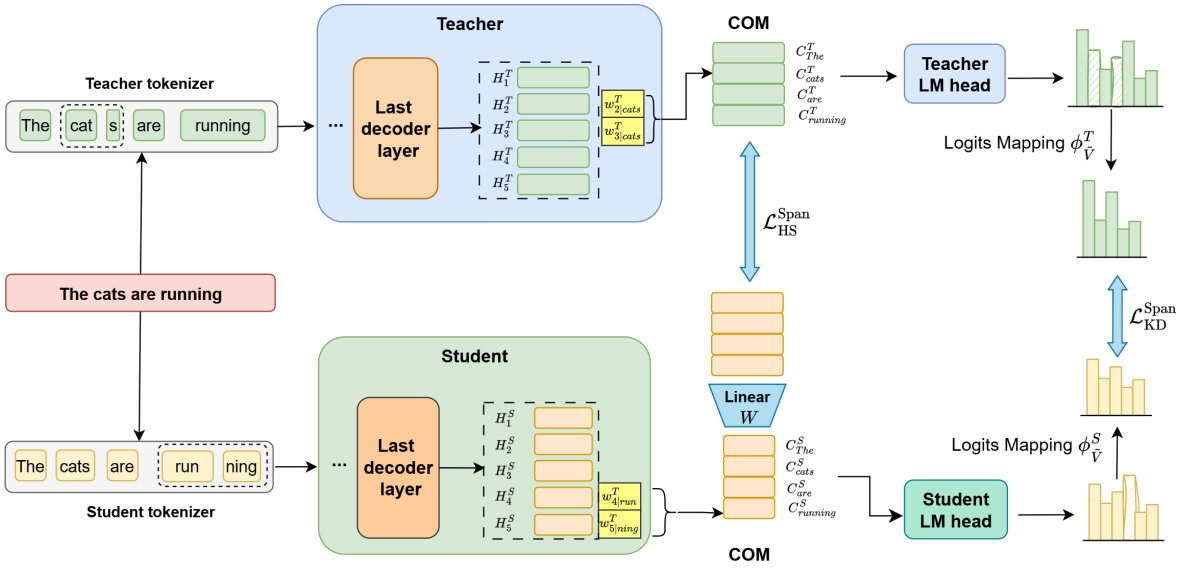

Figure 2: An illustration of the proposed SRA framework. Teacher–student spans are first matched using longest common subsequence (LCS). Span representations are then obtained via attention-weighted pooling. The student is guided to align its spans with those of the teacher through span-level hidden-state distillation loss ($\mathcal{L}^{Span}_{HS}$) and span-level logits distillation loss ($\mathcal{L}^{Span}_{KD}$).

i.e., $(t_1, \ldots, t_n)$ with offsets $(\text{off}_1, \ldots, \text{off}_n)$. Applying this process to both models yields two offset sequences $(\text{off}^T_1, \ldots, \text{off}^T_{n_T})$ and $(\text{off}^S_1, \ldots, \text{off}^S_{n_S})$ for the teacher and student, respectively. By computing the LCS of these offset sequences, we identify matched segments of the input that correspond to the same textual unit. These matched segments define the start and end indices of token spans in both models, thereby allowing us to extract pairs of aligned spans from the teacher and student representations. The detailed procedure is described in Appendix D.

We assign special tokens (e.g., [PAD]) an offset value 0 and then disregard them. Consequently, special tokens are not aligned. Compared to token-level alignment, span-level alignment ensures full coverage of the input sentence and mitigates the risk of information loss.

Span Representation via Weighted Token Pooling

While prior work has considered span-based distillation (Liu et al., 2022; Chen et al., 2025), most approaches adopt simple aggregation strategies such as mean pooling, which risk diluting the contribution of salient tokens. In contrast, we construct span representations through weighted aggregation of token embeddings, supported by a theoretical grounding that ensures more faithful preservation of semantic information.

The Transformer can be interpreted from a MPDS perspective (Section 2.2.2), tokens in a sentence correspond to particles moving in a d-dimensional space, and the hidden states at each layer represent their positions over time. The sequence of hidden states across layers then approximates the trajectories of these particles.

In physics, a center of mass serves as a rigorous representative of a cluster of particles, aggregating the positions and masses of all particles into a single point. A key property of this concept is its hierarchical nature: the center of mass of a larger cluster can be computed as a weighted average of the centers of mass of its subclusters or individual particles (see Appendix A). Inspired by the concept of the center of mass, we propose to model the representation of a text span (a cluster of tokens) analogously to the center of mass. This perspective motivates our definition of *a span representation as a linear combination of its constituent token or sub-span representations*. By defining our span representations possess this property, we ensure they can be computed hierarchically from smaller sub-spans.

Based on this principle, instead of aligning individual tokens (particle-level), we align spans (center-of-mass level), thereby transferring features at the span level (Section 3.2 and Section 3.3) to address the CTKD problem. This formulation constitutes the core idea of our approach, enabling the

model to capture both semantic and structural information while respecting the token-level structure. Since our knowledge transfer is performed on the last hidden states, we define the token-level weights at the final layer L as:

$$w_i = \frac{\tilde{w}_i}{\sum_{t=1}^{N} \tilde{w}_t}, \quad \tilde{w}_i = \sum_{a=1}^{A_h} \text{Att}_{a,N,i} \tag{2}$$

where A_h is the number of attention heads, N is the sequence length, and $\text{Att}_{a,N,i}$ denotes the attention score from last token N to token i in head a at layer L. As the last token in an LLM typically encodes the global contextual information of the entire sequence, $\tilde{w}_i$ quantifies the importance of token i via the attention it receives from the last token aggregated across all heads.

Given the aligned spans identified by $Ms = [(Span_0^S, Span_0^T), \ldots, (Span_n^S, Span_n^T)]$, with $Span_i^S = (s_i^S, e_i^S)$ and $Span_i^T = (s_i^T, e_i^T)$, where s and e denote the start and end token indices of a span, the span representations at the final layer for the student and teacher models are then computed as:

$$C_i^S = \sum_{t=s_i^S}^{e_i^S} w_t^S \, H_t^S \tag{3}$$

$$C_i^T = \sum_{t=s_i^T}^{e_i^T} w_t^T \, H_t^T \tag{4}$$

where H_t^S and H_t^T denote the last hidden representations of token t in the student and teacher, respectively.

3.2 Span-Level Hidden State Transfer

Under the lens of Transformers as a Multi-Particle Dynamical System, knowledge distillation can be viewed as transferring the salient characteristics of particles, where each particle's position corresponds to a token's hidden state. Consequently, distilling feature positions amounts to aligning the hidden states between teacher and student, consistent with prior work demonstrating the effectiveness of hidden-state distillation (Sanh et al., 2020; Jiao et al., 2019).

However, most existing methods treat all tokens as equal contributors to the distillation loss. From the perspective of the *center of mass* (CoM), this assumption is overly restrictive. In a particle system, each particle has a mass, and heavier particles pull the center of mass closer to themselves, reflecting their greater contribution to the global dynamics. Analogously, tokens vary in semantic importance: highly attended tokens act as "heavier" particles that should dominate the representation of the sentence, while less informative tokens contribute less.

Extending this analogy from tokens to spans, we argue that uniform weighting of spans is suboptimal, as it forces the student to overfit to irrelevant signals and dilutes the transfer of meaningful knowledge from teacher to student. Instead, we propose to assign attention-derived weights to spans, reflecting their semantic salience in the teacher model. Building on this intuition, we formulate a weighted hidden-state transfer loss, $\mathcal{L}_{\text{HS}}^{\text{Span}}$, that emphasizes the most informative spans while down-weighting unimportant ones:

$$\mathcal{L}_{\text{HS}}^{\text{Span}} = \sum_{i=1}^{N_{sp}} w_i^{\text{sp}} \, \mathcal{L}_{\cos}\big(C_i^S W, C_i^T\big) + \lambda \mathcal{L}_{\text{Geo}} \tag{5}$$

where N_{sp} is the number of aligned spans, $\mathcal{L}_{\cos}(u, v) = 1 - \frac{u \cdot v}{\|u\| \, \|v\|}$, C_i^S and C_i^T are the student and teacher representations of span i from the final layer (Eq. 3, Eq. 4), and W is a learnable projection matrix. The coefficient λ is a balancing hyperparameter that controls the strength of the geometric regularizer ($\mathcal{L}_{\text{Geo}}$).

Unconstrained linear projections can distort the geometry of representations and misalign their relative positions (Miles et al., 2024). To mitigate this issue, we introduce a geometric structure regularizer $\mathcal{L}_{\text{Geo}}$ that preserves relative geometry between spans:

$$\mathcal{L}_{\text{Geo}} = \sum_{i=1}^{N_{sp}} \sum_{j=i+1}^{N_{sp}} w_{i,j}^{\text{sc}} \big(d(C_i^S, C_j^S) - d(C_i^T, C_j^T)\big)^2 \tag{6}$$

with

$$w_{i,j}^{\text{sc}} = \frac{w_i^{\text{sp}} w_j^{\text{sp}}}{\sum_{i=1}^{N_{sp}} \sum_{j=i+1}^{N_{sp}} w_i^{\text{sp}} w_j^{\text{sp}}} \tag{7}$$

where $d(\cdot, \cdot)$ is the cosine function. The normalization ensures $\sum_{i<j} w_{i,j}^{\text{sc}} = 1$, making the loss comparable across samples. Preserving relative positions is essential for faithful knowledge transfer under the multi-particle perspective.

Reusing the teacher's token weight w_t^T (Eq. 2),

we define the normalized span weight as:

$$w_i^{\text{sp}} = \frac{\tilde{w}_i^{\text{sp}}}{\sum_{t=1}^{N_{sp}} \tilde{w}_t^{\text{sp}}}, \quad \tilde{w}_i^{\text{sp}} = \left(\sum_{t=s_i^T}^{e_i^T} w_t^T \right)^p \tag{8}$$

where s_i^T and e_i^T are the start and end token indices of the i-th teacher span. The hyperparameter p controls the sharpness of the weight distribution; when $p = 0$, all spans are weighted equally.

3.3 Span-Level Logits Transfer

We incorporate logits distillation into span-level knowledge transfer. Unlike previous approaches such as DSKD (Zhang et al., 2024b) or MINED (Wan et al., 2024) , which require complex token-level logits alignment, we take a multi-particle perspective. From this view, the logits of each token can be regarded as the position of a particle projected into the vocabulary space. Consequently, token-level logits can be naturally aggregated into span-level logits, enabling more faithful and robust knowledge transfer.

The primary challenge of vocabulary mismatch ($V_{\text{tea}} \neq V_{\text{stu}}$) is then interpreted as the teacher and student particles residing in distinct spaces. Our solution is to perform knowledge transfer within the shared subspace $\tilde{V} = V_{\text{tea}} \cap V_{\text{stu}}$. This shared subspace represents the overlapping dimensions where particle positions are directly comparable, allowing lexical knowledge to be meaningfully transferred. We formally define the aligned span logits as:

$$\tilde{y}^S_{1:N_{sp}} = \phi^S_{\tilde{V}} \left(f^S_{head}(C^S_{1:N_{sp}}) \right) \tag{9}$$

$$\tilde{y}^T_{1:N_{sp}} = \phi^T_{\tilde{V}} \left(f^T_{head}(C^T_{1:N_{sp}}) \right) \tag{10}$$

where $\phi^S_{\tilde{V}}(\cdot)$ and $\phi^T_{\tilde{V}}(\cdot)$ are mapping functions projecting the student and teacher logits into the shared subspace $\tilde{V}$, and f^S_{head}, f^T_{head} are the LM head layers.

Building on these definitions, we implement the aligned span logits loss, which aligns the distributions of the teacher's and student's aligned span logits. We then define the loss as:

$$\mathcal{L}^{\text{Span}}_{\text{KD}} = \sum_{i=1}^{N_{sp}} \mathcal{L}_{\text{KL}} \left(\tilde{y}_i^T, \tilde{y}_i^S; \tau \right) \tag{11}$$

where $\mathcal{L}_{\text{KL}}(\tilde{y}_i^T, \tilde{y}_i^S; \tau)$ denotes the Kullback–Leibler divergence between the teacher's and student's aligned span logits after applying a temperature-scaled (τ) softmax.

3.4 Combined Distillation Objective

The overall training objective in SRA integrates complementary span-level distillation signals. First, the span-level hidden state loss $\mathcal{L}^{\text{Span}}_{\text{HS}}$ encourages the student to imitate the teacher's span representations, which are weighted by semantic importance. This loss also incorporates a geometric structure regularizer, $\mathcal{L}_{\text{Geo}}$, to preserve the relative relationships between spans and prevent distortion of the learned space. Second, the aligned span logits loss $\mathcal{L}^{\text{Span}}_{\text{KD}}$ ensures knowledge transfer in the vocabulary space by aligning predictive distributions restricted to the shared subspace $\tilde{V}$. To further ensure effective learning, we balance these distillation signals with the task-specific supervision from ground-truth labels. Formally, the overall loss $\mathcal{L}_{\text{overall}}$ is defined as:

$$\mathcal{L}_{\text{overall}} = \alpha\, \mathcal{L}_{\text{CE}} + (1 - \alpha) \left(\mathcal{L}^{\text{Span}}_{\text{HS}} + \mathcal{L}^{\text{Span}}_{\text{KD}} \right) \tag{12}$$

where $\alpha \in [0, 1]$ balances the contribution of the standard cross-entropy loss $\mathcal{L}_{\text{CE}}$ and the span representation distillation loss.

4 Experiments

4.1 Experimental Setup

Datasets. Our evaluation spans multiple instruction-following datasets. We adopt the preprocessing procedure of (Wan et al., 2024). Distillation is trained on DATABRICKS-DOLLY-15K. For evaluation, we report ROUGE-L on the Dolly test split and on four benchmarks - S-NI (Wang et al., 2022), VICUNAEVAL (Chiang et al., 2023), DIALOGSUM (Chen et al., 2021), and SELFINST (Wang et al., 2023) - providing a broad view of models' cross-domain generalization.

Training and Evaluation Settings. Our experiments target cross-tokenizer distillation where teacher and student use different vocabularies. Students span GPT-2 (120M, 340M, 1.5B) (Radford et al., 2019), TinyLLaMA-1.1B (Zhang et al., 2024a), and OPT-2.7B (Zhang et al., 2022). We evaluate the following teacher→student pairs: Qwen1.5–1.8B (Bai et al., 2023) → GPT-2-120M / GPT-2-340M; Qwen2.5-7B-Instruct (Yang et al., 2024) → GPT-2-1.5B / OPT-2.7B; Mistral-7B (Jiang et al., 2023)→ TinyLLaMA-1.1B; and GPT-2-1.5B → GPT-2-120M. Training and evaluation setup, and details of baseline models are provided

in Appendix B. Reported evaluation results are averaged over five random seeds.

Baselines. We compare **SRA** with current CTKD methods [1]: ULD (Boizard et al., 2025), DSKD (Zhang et al., 2024b), MinED (Wan et al., 2024) and MultilevelOT (Cui et al., 2025).

4.2 Main Results

Table 1 provides an overview of the ROUGE-L evaluation results for all teacher–student configurations examined in this study. The first two sections correspond to medium-size teacher–student pairs, while the last three sections summarize the results for larger models. Across all teacher–student pairs and evaluation benchmarks, SRA consistently outperforms strong CTKD baselines (ULD, MinED, DSKD, and MultilevelOT), achieving the highest scores on most datasets and the best average ROUGE-L in every setting. These results highlight the effectiveness of span-level alignment in narrowing the performance gap between student and teacher models. We additionally compare SRA against ALM (Minixhofer et al., 2025), a span-based cross-tokenizer distillation method; as shown in Appendix C.4, SRA consistently outperforms ALM across all benchmarks and representative teacher–student configurations.

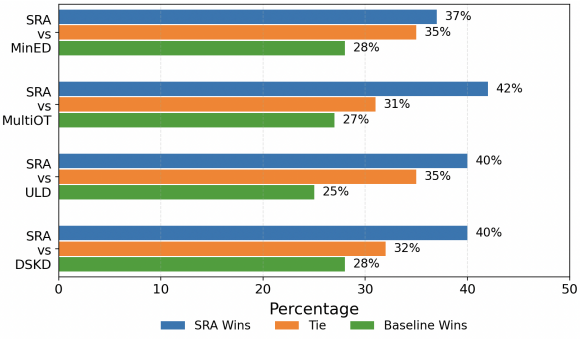

Figure 3: Win rates (%) for distilling Qwen 2.5-7B→GPT2 1.5B, evaluated by GPT-4o-mini.

Figure 3 reports the results of the semantic evaluation. To compare SRA with each baseline, we randomly sampled data from the combined benchmark corpus and employed `GPT-4o-mini` as an automatic judge. The model was prompted to determine which system's output was semantically superior by considering multiple criteria, including helpfulness, relevance and accuracy when responding to diverse instructions. Complementing the quantitative results in the previous table, these findings further confirm that SRA consistently delivers more semantically faithful and contextually aligned responses, demonstrating its strong robustness and

Methods	Dolly	Vicuna	SelfInst	S-NI	Dialog	Avg.
Qwen1.5–1.8B → GPT-2 120M						
Teacher	28.23	19.59	19.58	34.36	14.18	23.19
SFT	23.78	17.04	6.78	7.81	8.29	12.74
ULD	23.77	14.33	9.30	14.04	8.63	14.01
MinED	24.21	14.96	10.02	16.40	9.79	15.08
MultiLevelOT	23.02	13.79	8.41	12.26	8.79	13.25
DSKD	**24.26**	15.25	10.07	17.15	10.03	15.35
SRA	23.36	**16.08**	**13.16**	**23.70**	**13.56**	**17.97**
Qwen1.5–1.8B → GPT-2 340M						
Teacher	28.23	19.59	19.58	34.36	14.18	23.19
SFT	23.11	14.89	9.09	13.03	8.00	13.62
ULD	23.90	15.04	9.96	16.26	8.76	14.78
MinED	24.48	15.56	11.21	15.69	8.98	15.18
MultiLevelOT	23.95	14.80	10.21	15.87	8.99	14.76
DSKD	**25.43**	15.08	11.29	17.18	8.90	15.57
SRA	23.97	**16.31**	**13.64**	**24.49**	**12.08**	**18.10**
Qwen2.5-7B-Instruct → GPT2-1.5B						
Teacher	28.49	20.48	24.67	39.87	16.86	26.07
SFT	21.83	15.95	13.62	21.66	10.91	16.79
ULD	24.52	15.94	15.11	26.18	11.72	18.69
MinED	25.52	16.15	15.39	26.25	11.79	19.02
MultilevelOT	24.40	15.97	14.53	23.94	10.84	17.94
DSKD	25.38	16.84	16.10	25.82	12.19	19.27
SRA	**26.45**	**18.25**	**17.22**	**29.50**	**13.52**	**20.99**
Qwen2.5-7B-Instruct → OPT-2.7B						
Teacher	28.49	20.48	24.67	39.87	16.86	26.07
SFT	27.10	16.60	13.90	24.90	10.62	18.62
ULD	26.65	16.97	15.37	25.44	12.15	19.32
MinED	26.89	17.04	14.98	25.94	11.78	19.33
MultilevelOT	26.76	16.56	15.51	24.84	11.43	19.02
DSKD	26.93	17.86	16.22	27.33	12.43	20.15
SRA	**28.52**	**18.48**	**17.14**	**27.35**	**13.13**	**20.92**
Mistral-7B → TinyLLaMA-1.1B						
Teacher	32.15	20.43	25.44	36.88	14.67	25.91
SFT	23.20	15.70	15.70	28.43	10.77	18.76
ULD	25.48	17.31	17.72	32.54	11.75	20.96
MinED	25.54	17.02	18.23	31.42	11.77	20.80
MultilevelOT	24.56	16.84	15.61	27.91	12.04	19.40
DSKD	**26.28**	18.74	17.19	31.93	12.53	21.33
SRA	25.02	**19.69**	**20.05**	**32.98**	**14.88**	**22.52**

Table 1: Performance comparison of different teacher–student model combinations for CTKD.

[1]We exclude CDM (Chen et al., 2025) from our baselines. Although we attempted to implement it, the per-step computational cost—driven by its dynamic programming alignment for each training batch (as shown in Table 5)—proved to be substantially more expensive and impractical for our large-scale experimental setup.

generalization capability across different tasks.

5 Analysis

This section further demonstrates the contribution of each component in our method. Additional ablation studies are provided in Appendix C.

Impact of Span-Level Hidden State Transfer Table 2 shows that transferring the last-layer hidden state knowledge enhances the generative ability of the student models. Both the geometric loss ($\mathcal{L}_{\text{Geo}}$) and the cosine loss ($\mathcal{L}_{\text{cos}}$) contribute individual gains. The geometric loss gives the largest gain, and the cosine loss, though minor, further improves results when combined. This confirms the effectiveness of SRA's design, where complementary loss terms jointly strengthen cross-tokenizer knowledge transfer.

Ablation on Weighting Mechanisms Table 3 further isolates the effect of Weighted Span Pooling (WSP) and Weighted Span Loss (WSL). Removing WSP is equivalent to applying mean pooling over spans, while ablating WSL treats all tokens as equally important in the span-level objective. These components play a crucial role in forming high-quality span representations prior to execution of the method. The results show that removing either term weakens span representations and leads to less effective knowledge transfer, ultimately degrading distillation performance. This highlights the importance of well-structured span-level hidden state transfer in the overall effectiveness of SRA.

Results with shared vocabulary Table 4 demonstrates that SRA outperforms two categories of existing methods. The first includes distance-based techniques for shared-tokenizer scenarios, such as SeqKD (Kim and Rush, 2016), reverse KL (RKL) and Jensen–Shannon (JS) divergence (Wen et al., 2023), skewed KL (SKL) and skewed reverse KL (SRKL) divergence (Ko et al., 2024), and adaptive KL divergence (AKL) (Wu et al., 2024). The second includes methods addressing tokenizer discrepancies, such as ULD (Boizard et al., 2025), MinED (Wan et al., 2024), DSKD (Zhang et al., 2024b), and MultilevelOT (Cui et al., 2025). SRA yields higher ROUGE-L scores across multiple datasets and the highest overall average, despite being primarily designed to handle tokenizer discrepancies. This improvement can stem from SRA's ability to capture diverse representational differences between teacher and student, leading to more stable and comprehensive knowledge transfer.

$\mathcal{L}_{KD}^{Span}$	$\mathcal{L}_{Geo}$	$\mathcal{L}_{cos}$	Dolly	Vicuna	SelfInst	S-NI	Dialog	Avg
Qwen1.5–1.8B → GPT-2 340M								
✓			22.98	**16.32**	11.76	24.10	11.64	17.36
✓	✓		23.58	16.05	**13.73**	**25.02**	11.32	17.94
✓		✓	23.09	16.23	12.76	24.78	10.84	17.54
	✓	✓	**25.56**	15.93	13.28	23.70	8.95	17.48
✓	✓	✓	23.97	16.31	13.64	24.49	**12.08**	**18.10**
Qwen1.5–1.8B → GPT-2 120M								
✓			**23.74**	15.85	12.03	22.11	11.78	17.10
✓	✓		22.80	15.96	12.94	**25.05**	11.85	17.72
✓		✓	23.20	**16.16**	12.02	22.55	12.69	17.32
	✓	✓	23.24	15.18	11.02	21.45	9.32	16.04
✓	✓	✓	23.36	16.08	**13.16**	23.70	**13.56**	**17.97**

Table 2: Ablation study on loss components across two teacher–student pairs. **Bold** denotes the highest score and underline the second highest.

WSL	WSP	Dolly	Vicuna	SelfInst	S-NI	Dialog	Avg
Qwen1.5-1.8B → GPT-2 340M							
		22.49	14.25	12.90	23.30	12.02	16.99
	✓	23.18	15.57	12.82	21.99	12.01	17.11
✓		23.25	15.43	12.82	22.35	**12.93**	17.36
✓	✓	**23.97**	**16.31**	**13.64**	**24.49**	12.08	**18.10**
Qwen1.5-1.8B → GPT-2 120M							
		20.12	12.23	11.50	19.31	11.07	14.85
	✓	20.86	13.16	12.36	20.97	11.52	15.77
✓		21.54	13.49	11.09	20.69	12.62	15.89
✓	✓	**23.36**	**16.08**	**13.16**	**23.70**	**13.56**	**17.97**

Table 3: Ablations on weighting mechanisms. Weighted Span Loss (WSL); Weighted Span Pooling (WSP).

Methods	Dolly	Vicuna	SelfInst	S-NI	Avg.
GPT-2-1.5B → GPT-2–120M					
Teacher	27.19	16.30	14.64	27.55	21.42
SFT	22.94	15.17	10.11	16.21	16.11
SeqKD	23.68	14.41	10.03	16.61	16.18
RKL	24.34	15.71	10.53	17.31	17.03
JS	23.86	15.50	10.20	16.20	16.44
SKL	24.03	14.70	10.66	17.99	16.85
SRKL	23.48	14.91	10.35	16.53	16.32
AKL	**24.75**	15.37	10.46	17.48	17.02
ULD	23.53	14.89	10.47	15.43	16.08
MinED	23.69	15.17	10.43	15.84	16.28
MultiLevelOT	23.81	14.91	10.70	14.91	16.08
DSKD	23.93	15.00	10.66	16.81	16.60
SRA	23.21	**16.17**	**12.53**	**25.06**	**19.24**

Table 4: Experiment on same tokenizer distillation.

Training Efficiency Table 5 reports the training efficiency of **SRA** compared to CTKD baselines under identical settings. Although SRA requires

Method	avg_alloc (GB)	avg_step_time (s)
CDM†	22.61	1.0100
DSKD	20.11	0.3520
MinED	19.63	0.4244
ULD	19.63	0.4393
SRA	21.96	**0.2754**

Table 5: Average GPU memory allocation and per-step training time for Qwen2.5-7B-Instruct → GPT2-1.5B on a single A100 40GB GPU (200 steps). DSKD, MinED, ULD, and SRA use batch size = 4; †CDM is run with batch size = 1 due to memory constraints.

slightly more GPU memory than MinED and ULD (≈1–1.5 GB on a 40 GB card), owing to the additional span matching and attention-based aggregation components, it achieves the **lowest average per-step training time** among all methods. In other words, the span-level geometric terms introduce only a modest memory increase while keeping the overall training cost comparable to, and in this setting even more time-efficient than, strong CTKD baselines.

Robustness to Vocabulary Mismatch Table 6 reports the vocabulary overlap between each teacher–student pair used in our experiments. In all configurations, the shared vocabulary covers 76–84% of the student's vocabulary, ensuring that span-level logit distillation operates over a substantial and semantically meaningful subset of the lexical space. Notably, even without logit distillation loss (L_{KD}^{Span}), the span-level hidden-state alignment alone ($\mathcal{L}_{cos}$ + $\mathcal{L}_{Geo}$) already yields competitive performance against strong CTKD baselines (see in Appendix C.3), demonstrating that our core span representations are inherently tokenizer-agnostic. However, incorporating the logit distillation loss $\mathcal{L}_{cos}$ over the shared vocabulary further boosts performance across all configurations (see Table 2).

Teacher	Student	$\lvert V_S \rvert$	$\lvert V_T \rvert$	$\lvert V_S \cap V_T \rvert$	%
Qwen1.5-1.8B	GPT-2 120M	50,257	151,646	42,257	84
Qwen1.5-1.8B	GPT-2 340M	50,257	151,646	42,257	84
Qwen2.5-7B	GPT2-1.5B	50,257	151,665	42,257	84
Qwen2.5-7B	OPT-2.7B	50,265	151,665	42,260	84
Mistral-7B	TinyLLaMA-1.1B	32,000	32,000	24,184	76

Table 6: Vocabulary overlap between teacher and student tokenizers. % denotes the fraction of the student vocabulary present in the teacher vocabulary.

6 Conclusion

We introduced **SRA**, a framework for CTKD that leverages span-level alignment to bridge the gap between heterogeneous tokenizations. By aligning sequences via LCS, SRA provides a principled way to construct comparable span representations while capturing span importance. Extensive experiments across multiple benchmarks demonstrate that SRA consistently outperforms most advanced distillation methods. Our work provides a robust framework for practical knowledge transfer. Future research could extend SRA to embedding model scenarios or other representation learning settings.

7 Limitations

Our work was conducted under limited computational budgets, which constrained the scope of experimentation. Moreover, the current logit mapping in our framework is static. While this design proved effective, it inevitably overlooks other important dimensions of the logit space that may carry valuable knowledge. Furthermore, aligning the span representation spaces necessitates on-the-fly teacher inference, since precomputing all teacher span embeddings would require enormous storage. Finally, our experiments were conducted under fixed computational budgets and benchmark conditions. We view these limitations as opportunities for further development, particularly in designing more efficient span-level alignment mechanisms and adaptive mapping strategies that can make SRA both more scalable and more reliable.

Acknowledgments

This project was supported by the Air Force Office of Scientific Research under award number FA9550-23-S-0001.

References

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang

Zhu. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Nicolas Boizard, Kevin El Haddad, Céline Hudelot, and Pierre Colombo. 2025. Towards cross-tokenizer distillation: the universal logit distillation loss for llms.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David K. Duvenaud. 2018. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31.

Yijie Chen, Yijin Liu, Fandong Meng, Yufeng Chen, Jinan Xu, and Jie Zhou. 2025. Enhancing cross-tokenizer knowledge distillation with contextual dynamical mapping.

Yulong Chen, Yang Liu, Liang Chen, and Yue Zhang. 2021. DialogSum: A real-life scenario dialogue summarization dataset. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 5062–5074, Online. Association for Computational Linguistics.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing GPT-4 with 90%* ChatGPT quality.

Xiao Cui, Mo Zhu, Yulei Qin, Liang Xie, Wengang Zhou, and Houqiang Li. 2025. Multi-level optimal transport for universal cross-tokenizer knowledge distillation on language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 23724–23732.

DeepSeek-AI-Group. 2024. Deepseek-v3 technical report.

Guoqiang Gong, Jiaxing Wang, Jin Xu, Deping Xiang, Zicheng Zhang, Leqi Shen, Yifeng Zhang, JunhuaShu JunhuaShu, ZhaolongXing ZhaolongXing, Zhen Chen, et al. 2025. Beyond logits: Aligning feature dynamics for effective knowledge distillation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 23067–23077.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. 2024. Minillm: Knowledge distillation of large language models.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network.

Fengqing Jiang, Zhangchen Xu, Luyao Niu, Boxin Wang, Jinyuan Jia, Bo Li, and Radha Poovendran. 2023. Identifying and mitigating vulnerabilities in LLM-Integrated applications. *arXiv preprint arXiv:2311.16153*.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2019. Tinybert: Distilling BERT for natural language understanding. *CoRR*, abs/1909.10351.

Yoon Kim and Alexander M. Rush. 2016. Sequence-level knowledge distillation.

Jongwoo Ko, Sungnyun Kim, Tianyi Chen, and Se-Young Yun. 2024. Distillm: Towards streamlined distillation for large language models.

Tue Le, Hoang Tran Vuong, Quyen Tran, Linh Ngo Van, Mehrtash Harandi, and Trung Le. 2025. Token-level self-play with importance-aware guidance for large language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Chang Liu, Chongyang Tao, Jiazhan Feng, and Dongyan Zhao. 2022. Multi-granularity structural knowledge distillation for language model compression. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1001–1011.

Yiping Lu, Zhuohan Li, Di He, Zhiqing Sun, Bin Dong, Tao Qin, Liwei Wang, and Tie-Yan Liu. 2019. Understanding and improving transformer from a multi-particle dynamic system point of view. *arXiv preprint arXiv:1906.02762*.

Roy Miles, Ismail Elezi, and Jiankang Deng. 2024. Vkd: Improving knowledge distillation using orthogonal projections.

Benjamin Minixhofer, Ivan Vulić, and Edoardo Maria Ponti. 2025. Universal cross-tokenizer distillation via approximate likelihood matching.

Forest Ray Moulton. 1970. *An introduction to celestial mechanics*. Courier Corporation.

Truong Nguyen, Phi Van Dat, Ngan Nguyen, Linh Ngo Van, Trung Le, and Thanh Hong Nguyen. 2026. CTPD: cross tokenizer preference distillation. In *Fortieth AAAI Conference on Artificial Intelligence, Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence, Sixteenth Symposium on Educational Advances in Artificial Intelligence, AAAI*, pages 37783–37790. AAAI Press.

OpenAI. 2024. Gpt-4 technical report.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2020. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter.

Siqi Sun, Yu Cheng, Zhe Gan, and Jingjing Liu. 2019. Patient knowledge distillation for bert model compression.

Minh-Phuc Truong, Hai An Vu, Tu Vu, Nguyen Thi Ngoc Diep, Linh Ngo Van, Thien Huu Nguyen, and Trung Le. 2025. Emo: Embedding model distillation via intra-model relation and optimal transport alignments. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 7605–7617.

Duc Trung Vu, Pham Khanh Chi, Dat Phi Van, Linh Ngo Van, Dinh Viet Sang, and Trung Le. 2026a. Dwa-kd: Dual-space weighting and time-warped alignment for cross-tokenizer knowledge distillation. In *Findings of the Association for Computational Linguistics: EACL*, pages 3513–3527.

Hai An Vu, Minh-Phuc Truong, Tu Vu, and Linh Ngo. 2026b. Mol: Mixture of layers in cross-tokenizer embedding model distillation. *Knowledge-Based Systems*, 343:116001.

Hoang Tran Vuong, Tue Le, Quyen Tran, Linh Ngo Van, and Trung Le. 2026. MCW-KD: multi-cost wasserstein knowledge distillation for large language models. In *Fortieth AAAI Conference on Artificial Intelligence, Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence, Sixteenth Symposium on Educational Advances in Artificial Intelligence, AAAI*, pages 33332–33340. AAAI Press.

Fanqi Wan, Xinting Huang, Deng Cai, Xiaojun Quan, Wei Bi, and Shuming Shi. 2024. Knowledge fusion of large language models.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. Self-Instruct: Aligning language models with Self-Generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508, Toronto, Canada. Association for Computational Linguistics.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Gary Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Maitreya Patel, Kuntal Kumar Pal, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Shailaja Keyur Sampat, Savan Doshi, Siddhartha Mishra, Sujan Reddy, Sumanta Patro, Tanay Dixit, Xudong Shen, Chitta Baral, Yejin Choi, Noah A. Smith, Hannaneh Hajishirzi, and Daniel Khashabi. 2022. Super-naturalinstructions: Generalization via declarative instructions on 1600+ NLP tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5085–5109, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yuqiao Wen, Zichao Li, Wenyu Du, and Lili Mou. 2023. f-divergence minimization for sequence-level knowledge distillation.

Taiqiang Wu, Chaofan Tao, Jiahao Wang, Runming Yang, Zhe Zhao, and Ngai Wong. 2024. Rethinking kullback-leibler divergence in knowledge distillation for large language models.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Peiyuan Zhang, Guangtao Zeng, Tianduo Wang, and Wei Lu. 2024a. TinyLlama: An open-source small language model. *arXiv preprint arXiv:2401.02385v2*.

Songming Zhang, Xue Zhang, Zengkui Sun, Yufeng Chen, and Jinan Xu. 2024b. Dual-space knowledge distillation for large language models.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. OPT: Open pre-trained transformer language models.

A Center-of-Mass Dynamics and Transformer Analogy

Multi-Particle Dynamic Systems (MPDS) model the motion of particle collections in space using differential equations (Moulton, 1970). Each particle's behavior is influenced by convection, governing its intrinsic dynamics, and diffusion, capturing interactions with other particles.The Transformer architecture can be interpreted through the lens of a multi-particle dynamic system (Lu et al., 2019).

In physics, consider a system of N particles, where each particle has mass m_i and position x_i. The center of mass of the system is given by:

$$\tilde{x}(t) = \frac{1}{M}\sum_{i=1}^{N} m_i(t)x_i(t), \quad M = \sum_{i=1}^{N} m_i(t)$$

These expressions extend naturally to the case where the N particles are partitioned into K clusters:

$$\begin{aligned} \tilde{x}(t) &= \frac{1}{M}\sum_{k=1}^{K} M_k(t)\,\tilde{x}_k(t), \\ \tilde{x}_k(t) &= \frac{1}{M_k(t)}\sum_{i\in C_k} m_i(t)\,x_i(t), \\ M_k(t) &= \sum_{i\in C_k} m_i(t). \end{aligned} \tag{13}$$

Thus, both the center-of-mass can be computed hierarchically through sub-clusters.

Analogy to Transformers. We can establish an analogy between particle dynamics and the evolution of hidden states across Transformer layers. Let l denote the layer index, and identify $x_i(t)$ with hidden states $H_{l,i}$. The center-of-mass position of span k at layer l is given by:

$$\begin{aligned} (H_l)_k^{\text{Span}} &= \frac{1}{M_{l,k}}\sum_{i\in \text{Span}_k} w_{l,i}\,H_{l,i}, \\ M_{l,k} &= \sum_{i\in \text{Span}_k} w_{l,i}. \end{aligned} \tag{14}$$

where $(H_l)^{\text{Span}}k$ denotes the span representation, and wl, i plays the role of the "mass" assigned to token i at layer l.

Hence, by defining span representations as center-of-mass positions, their statescan be computed hierarchically—analogous to clusters of particles. This perspective enables us to exploit well-established physical principles to design a systematic and physically grounded mechanism for knowledge transfer.

B Experimental Details

Training and Evaluation For GPT2-120M and GPT2-340M, we employ full fine-tuning. For TinyLLaMA, GPT2-1.5B, and OPT2.7B, we adopt LoRA-based fine-tuning. Detailed training configurations for each model are summarized in Table 8. All knowledge distillation (KD) experiments are performed on the Databricks-Dolly-15k dataset. We evaluate across multiple datasets covering different domains and tasks, using validation ROUGE-L (Lin, 2004) for model selection.

Hyperparameter The hyperparameters are reported in Table 9. Throughout all experiments, we set the KL-divergence temperature τ to 2.0, hyperparameter p to 1.0 and the weight λ of the geometric regularization constraint to 50.

C Additional Ablation Results

C.1 Effect of Hyperparameter p

The results in 10 demonstrate that applying weighting through the sharpness parameter p consistently improves model performance, with the best average achieved at $p = 0.5$. This suggests that moderate weighting helps the student model better align with the teacher's distribution. However, further increasing p to 1.0 brings no improvement, suggesting a weighting threshold beyond which the model no longer benefits from sharper distributions.

C.2 Effect of Transferred Layers

Increasing the number of transferred layers consistently reduces performance (Table 11). We hypothesize that transferring from too many intermediate layers amplifies architectural and representational mismatches between the teacher (Qwen1.5) and the student (GPT-2), leading to unstable supervision and degraded knowledge transfer efficiency.

C.3 The effect of Span-Level Hidden State Transfer without Span-Level Logits loss

Table 12 reports the performance of SRA with and without the span-level logit distillation loss $\mathcal{L}_{KD}^{Span}$ across two teacher–student pairs. Several observations are noteworthy. First, even without $\mathcal{L}_{KD}^{Span}$, SRA already surpasses all CTKD baselines in average ROUGE-L (16.04 vs. 15.35 for GPT-2 120M;

Transformer Concept	$\leftrightarrow$ **Multiparticle Dynamical Analog**
Token Hidden State $h_{\ell,i}$	$\leftrightarrow$ Particle Position $x_{i,t} \in \mathbb{R}^d$
Layer Index ℓ	$\leftrightarrow$ Discretized Time Step t
Multi-Head Self-Attention $\mathrm{MHA}_{W^{\ell}_{\text{att}}}(h_{\ell,i}, [h_{\ell,1}, \ldots, h_{\ell,n}])$	$\leftrightarrow$ **Diffusion:** inter-particle interaction $F\big(x_{i,t}, [x_{1,t} \ldots, x_{n,t}]\big)$
Feed-Forward Network (per token) $\mathrm{FFN}_{W^{\ell}_{\text{ffn}}}(\tilde{h}_{\ell,i})$	$\leftrightarrow$ **Convection:** private particle update $G(\tilde{x}_{i,t})$
Transformer Residual Update $\tilde{h}_{\ell,i} = h_{\ell,i} + \mathrm{MHA}_{W^{\ell}_{\text{att}}}\big(h_{\ell,i}, [h_{\ell,1}, \ldots, h_{\ell,n}]\big)$ $h_{\ell+1,i} = \tilde{h}_{\ell,i} + \mathrm{FFN}_{W^{\ell}_{\text{ffn}}}\big(\tilde{h}_{\ell,i}\big)$	$\leftrightarrow$ Lie–Trotter splitting step $\tilde{x}_i = x_i + \gamma F(x_{i,t}[x_{1,t} \ldots, x_{n,t}])$ $x_i^+ = \tilde{x}_i + \gamma\, G\big(\tilde{x}_{i,t}\big)$

Table 7: Dictionary aligning Transformer operations with their Multi-Particle Dynamical System (MPDS) counterparts (Lu et al., 2019).

Student	**Epoch**	**LR**	**Tuning**	**LoRA Dropout**	**LoRA Rank/Alpha**
Teacher: Qwen1.5-1.8B					
GPT2-120M	20	5e-4	Full	–	–
GPT2-340M	20	5e-4	Full	–	–
Teacher: Mistral-7B					
TinyLLaMA	15	1e-3	LoRA	0.1	16/64
Teacher: Qwen2.5-7B-Instruct					
GPT2-1.5B	15	1e-3	LoRA	0.1	16/64
OPT-2.7B	15	1e-3	LoRA	0.1	16/64

Table 8: Training configurations. Projector LR = 5e-4, batch size = 8, and cosine LR scheduler are shared across all settings.

17.48 vs. 15.57 for GPT-2 340M), demonstrating that the span-level hidden-state alignment and geometric regularization alone constitute a strong and tokenizer-agnostic distillation signal. Second, incorporating $\mathcal{L}_{KD}^{Span}$ consistently yields further gains across nearly all benchmarks, with the most pronounced improvements on Dialog (+4.24 for GPT-2 120M; +3.13 for GPT-2 340M) and SelfInst, confirming that logit distillation over the shared vocabulary provides a complementary supervisory signal that reinforces the teacher's predictive behavior.

GPT2-120M	**GPT2-340M**	**TinyLLaMA**	**GPT2-1.5B**	**OPT2.7B**
0.5	0.5	0.6	0.6	0.8

Table 9: The hyperparameters α for different configurations

Hyperparameter p	Teacher: Qwen1.5-1.8B $\rightarrow$ GPT-2 120M					
	Dolly	**Vicuna**	**Self Inst**	**S-NI**	**Dialog**	**Avg.**
$p = 0.0$	21.54	13.49	11.09	20.69	12.62	15.89
$p = 0.5$	**23.75**	15.52	12.92	**25.08**	12.66	**17.99**
$p = 1.0$	23.36	**16.08**	**13.16**	23.70	**13.56**	17.97

Table 10: Effect of the sharpness hyperparameter p on model performance.

Transferred Layers	Teacher: Qwen1.5-1.8B $\rightarrow$ GPT-2 120M					
	Dolly	**Vicuna**	**Self Inst**	**S-NI**	**Dialog**	**Avg.**
0	**23.74**	15.85	12.03	22.11	11.78	17.10
1 (12)	23.36	**16.08**	**13.16**	**23.70**	**13.56**	**17.97**
3 (10, 11, 12)	21.13	13.61	11.61	20.28	10.64	15.45
5 (8, 9, 10, 11, 12)	20.78	12.27	10.81	20.47	11.67	15.20

Table 11: Effect of the number of transferred layers on model performance.

These results collectively validate the design choice of combining tokenizer-agnostic span representations with vocabulary-intersection logit distillation in the full SRA objective.

C.4 Comparison with ALM

As shown in Table 13, **SRA** consistently outperforms ALM across all benchmarks and both teacher–student configurations. While ALM aggregates over spans at the token-likelihood level, it still relies on token-level predictive traces that are sensitive to tokenizer discrepancies. In contrast, SRA operates directly on span-level hidden states

Methods	Dolly	Vicuna	SelfInst	S-NI	Dialog	Avg.
Qwen1.5–1.8B → GPT-2 120M						
Teacher	28.23	19.59	19.58	34.36	14.18	23.19
SFT	23.78	17.04	6.78	7.81	8.29	12.74
ULD	23.77	14.33	9.30	14.04	8.63	14.01
MinED	24.21	14.96	10.02	16.40	9.79	15.08
MultiLevelOT	23.02	13.79	8.41	12.26	8.79	13.25
DSKD	**24.26**	15.25	10.07	17.15	10.03	15.35
SRA w/o $\mathcal{L}_{KD}^{Span}$	23.24	15.18	11.02	21.45	9.32	16.04
SRA	23.36	**16.08**	**13.16**	**23.70**	**13.56**	**17.97**
Qwen1.5–1.8B → GPT-2 340M						
Teacher	28.23	19.59	19.58	34.36	14.18	23.19
SFT	23.11	14.89	9.09	13.03	8.00	13.62
ULD	23.90	15.04	9.96	16.26	8.76	14.78
MinED	24.48	15.56	11.21	15.69	8.98	15.18
MultiLevelOT	23.95	14.80	10.21	15.87	8.99	14.76
DSKD	25.43	15.08	11.29	17.18	8.90	15.57
SRA w/o $\mathcal{L}_{KD}^{Span}$	**25.56**	15.93	13.28	23.70	8.95	17.48
SRA	23.97	**16.31**	**13.64**	**24.49**	**12.08**	**18.10**

Table 12: ROUGE-L comparison of SRA and SRA w/o $\mathcal{L}_{KD}^{Span}$ against CTKD baselines.

Methods	Dolly	Vicuna	SelfInst	S-NI	Dialog	Avg.
Qwen1.5–1.8B → GPT-2 120M						
Teacher	28.23	19.59	19.58	34.36	14.18	23.19
SFT	23.78	17.04	6.78	7.81	8.29	12.74
ALM	20.86	14.76	10.65	17.76	10.44	14.89
SRA	**23.36**	**16.08**	**13.16**	**23.70**	**13.56**	**17.97**
Qwen2.5-7B-Instruct → GPT2-1.5B						
Teacher	28.49	20.48	24.67	39.87	16.86	26.07
SFT	21.83	15.95	13.62	21.66	10.91	16.79
ALM	25.78	16.36	15.63	25.78	11.07	18.92
SRA	**26.45**	**18.25**	**17.22**	**29.50**	**13.52**	**20.99**

Table 13: ROUGE-L comparison of SRA against ALM across representative teacher–student pairs. SRA outperforms ALM in all settings.

with geometric regularization, bypassing token-level misalignment entirely. This design difference is reflected in the results: SRA achieves an average ROUGE-L gain of **+3.08** over ALM on the Qwen1.5–1.8B → GPT-2 120M pair and **+2.07** on the Qwen2.5-7B-Instruct → GPT2-1.5B pair, with pronounced improvements on benchmarks such as S-NI and Dialog, suggesting that span-level hidden-state alignment generalizes more robustly across instruction-following tasks.

D Longest Common Subsequence-based Span Alignment

Algorithm 1: LCS-based Span Alignment

Input:
Teacher offsets O^T,
Student offsets O^S,
First non-special token indices t_0^T, t_0^S

Output: List of matched span indices Ms

$m \leftarrow \text{len}(O^T)$, $n \leftarrow \text{len}(O^S)$;
$i \leftarrow 0$, $j \leftarrow 0$;
$aligns \leftarrow [(t_0^T, t_0^S)]$, $Ms \leftarrow [\,]$;
while $i < m$ ***and*** $j < n$ **do**
  if $O^T[i] = 0$ **then**
    $i \leftarrow i + 1$;
    continue;
  end
  if $O^S[j] = 0$ **then**
    $j \leftarrow j + 1$;
    continue;
  end
  if $O^T[i] = O^S[j]$ **then**
    $span^S = (aligns[-1][0], i)$;
    $span^T = (aligns[-1][1], j)$;
    append $(span^S, span^T)$ to Ms;
    append $(i + 1, j + 1)$ to $aligns$;
    $i \leftarrow i + 1,\ j \leftarrow j + 1$;
  else
    if $O^T[i] < O^S[j]$ **then**
      $i \leftarrow i + 1$;
    else
      $j \leftarrow j + 1$;
    end
  end
end
return Ms;

E Prompt for evaluation via GPT-4

Please act as an impartial judge and compare the quality of response A and response B provided by two AI assistants to the user question displayed below. Focus on how natural, fluent, and human-like the language sounds Your evaluation should prioritize effectiveness, clarity, readability, technical accuracy and completeness.

- If A is significantly better, answer "A".
- If B is significantly better, answer "B".
- If both are similar in quality (both bad or both good or show no significant difference), answer "Tied".

[Question]
{question or instruction}

[Response A]
{response A}

[Response B]
{response B}

Figure 4: Prompt for GPT-4 evaluation.