On-Policy Knowledge Distillation for Training Small Language Models

Presenter: Ngô Văn Linh

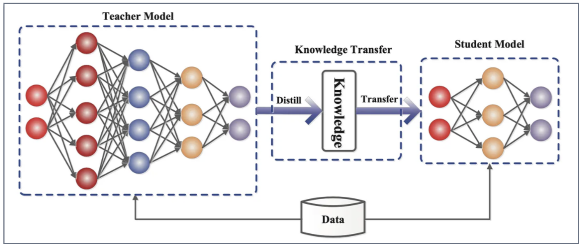
Outline

1. Introduction
2. Distillation Topics for LLMs
3. Selected Representative Papers
   - On-Policy Distillation of Language Models
   - Speculative Knowledge Distillation
   - DistiLLM: Towards Streamlined Distillation for Large Language Models
   - DistiLLM-2: A Contrastive Approach Boosts the Distillation of LLMs
   - Beyond Logits: Aligning Feature Dynamics for Effective Knowledge Distillation
   - MTA: Multi-Granular Trajectory Alignment for Large Language Model Distillation
   - TSD-KD: Improving Reasoning via Token-Selective Dual Knowledge Distillation
   - CSD: Distillation of Large Language Models via Concrete Score Matching
   - SRA: Span Representation Alignment
   - TALAS: Teacher-Anchored Layer Alignment
4. Conclusion
5. References

1. Introduction

- LLMs ngày càng lớn $\Rightarrow$ inference cost tăng, memory tăng.
- Knowledge Distillation (KD) [1] giúp nén mô hình:

Teacher large $\rightarrow$ Student smaller



Mô hình sinh chuỗi tự hồi quy

Mô hình sinh chuỗi tự hồi quy (Auto-regressive Generative Sequence Models)

- Gọi x là đầu vào, y là chuỗi đầu ra; $y_{<n+1} = (y_1, \ldots, y_n)$.
- Một policy tự hồi quy cấp token:

$$p(\cdot \mid y_{<n}, x) \in (0,1)^M$$

  cho phân phối xác suất token tiếp theo trên toàn bộ từ vựng M.
- Viết gọn $y \sim p(\cdot \mid x)$ để chỉ chuỗi y được sample từ model với đầu vào x, và

$$p(y_n \mid x) := p(y_n \mid y_{<n}, x).$$

Xác suất của token thứ n:

$$p(y_n \mid x) = \frac{\exp(z_n/\gamma)}{\sum_{i=1}^{M} \exp(z_i/\gamma)},$$

trong đó z_n là logit của token y_n.

KD cho Autoregressive Language Models (classic)

Mục tiêu: làm cho phân phối của student giống teacher bằng cách minimize divergence D giữa teacher $p(y \mid x)$ và student $q_\theta(y \mid x)$; ta dùng KL [2]:

- **KL trên phân phối câu**:

$$D_{\mathrm{KL}}(p, q_\theta) = \mathbb{E}_x \mathbb{E}_{y \sim p(\cdot|x)} \left[\log \frac{p(y \mid x)}{q_\theta(y \mid x)} \right]$$

- **Xấp xỉ bằng trung bình trên dataset $\mathcal{D}$**:

$$D_{\mathrm{KL}}(p, q_\theta) \approx \frac{1}{|\mathcal{D}|} \sum_{(x,y) \in \mathcal{D}} p(y \mid x) \log \frac{p(y \mid x)}{q_\theta(y \mid x)}$$

- **Tách thành tổng theo từng token**:

$$D_{\mathrm{KL}}(p, q_\theta) = \frac{1}{|\mathcal{D}_x|} \sum_{x \in \mathcal{D}_x} \sum_{t} \sum_{y_t \in V} p(y_t \mid y_{<t}, x) \log \frac{p(y_t \mid y_{<t}, x)}{q_\theta(y_t \mid y_{<t}, x)}$$

KL-Based Divergences

Reverse KL (KL ngược)

- Forward KL: $D_{KL}(P \parallel Q) = \sum_c P(c) \log \frac{P(c)}{Q(c)}$.
- **Reverse KL** hoán đổi vai trò của P và Q:

$$D_{KL}(Q \parallel P) = \sum_c Q(c) \log \frac{Q(c)}{P(c)}.$$

Jensen–Shannon divergence (JSD cơ bản)

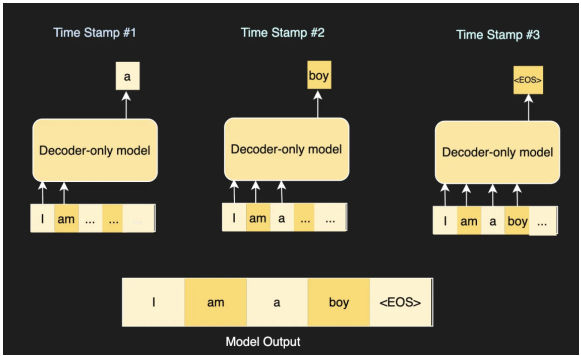
- Đặt phân phối trung gian: $M = \frac{1}{2}(P + Q)$.
- JSD định nghĩa:

$$D_{JS}(P \parallel Q) = \tfrac{1}{2} D_{KL}(P \parallel M) + \tfrac{1}{2} D_{KL}(Q \parallel M).$$

- JSD luôn hữu hạn và đối xứng, khác với KL.

Khó khăn

- Khó khăn khi distillation cho **auto-regressive LMs**:
  - Distribution mismatch giữa training data và student inference (Exposure Bias).
  - Model Capacity Mismatch và Structural incompatibility giữa teacher và student.



Exposure bias

Ý tưởng: Khi train (teacher forcing), model luôn nhìn token đúng ở bước trước. Khi suy luận, model phải dùng token do chính nó sinh ra $\Rightarrow$ chỉ một lỗi nhỏ ở sớm sẽ làm cả phân phối về sau lệch hẳn.

Training (teacher forcing)

$p_{\text{train}}(y_t \mid y_{<t} = \text{chuỗi } \textbf{đúng})$

Inference (student tự sinh)

$p_{\text{inf}}(y_t \mid y_{<t} = \text{chuỗi } \textbf{student})$

Tại training: model *luôn* thấy "Con mèo ngồi trên ghế"

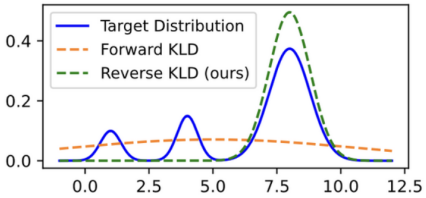
Tại inference: chỉ cần một lỗi "ngủ" thay vì "ngồi"

làm toàn bộ context về sau lệch hẳn.

Model capacity mismatch và Structural incompatibility

Model capacity mismatch: Teacher model phức tạp, nhiều modes, patterns, behaviours hơn khả năng biểu diễn của student model.

- Forward KL: $\mathrm{KL}(p\|q) = \mathbb{E}_p\left[\log \frac{p}{q}\right] \Rightarrow$ *mode covering/averaging*, gradient explodes when $q \to 0$.
- Reverse KL: $\mathrm{KL}(q\|p) \Rightarrow$ *mode seeking/collapse*. Student dễ bám vào vài mode lớn của teacher và bỏ qua mode nhỏ.



Structural incompatibility: Vấn đề cross-tokenizer, khác kiến trúc, số chiều biểu diễn.

Distillation Topics for LLMs

Từ model compression sang knowledge transfer

- Trọng tâm của KD cho LLM không còn chỉ là **nén mô hình**, mà là **trích xuất và chuyển giao tri thức hữu dụng** từ teacher sang student.
- Điều này đặc biệt quan trọng khi teacher là mô hình **độc quyền / black-box**.
- Vì vậy, field chuyển dần sang các tín hiệu giàu ngữ nghĩa hơn: **generated outputs, reasoning traces, preference signals**.

Ví dụ tiêu biểu

On-Policy Distillation of Language Models: Learning from Self-Generated Mistakes – Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos, Matthieu Geist, Olivier Bachem [3]

Zephyr: Direct Distillation of LM Alignment – Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M. Rush, Thomas Wolf [10]

Topic 1: response-based methods trở thành dòng chính

- Trong LLM distillation, ba nhánh chi phối là **logit distillation**, **sequence distillation**, và **preference distillation**.
- Điểm chung của chúng là đều bám vào **hành vi sinh đầu ra**, thay vì phụ thuộc mạnh vào biểu diễn trung gian.
- Nói cách khác, nghiên cứu KD cho LLM hiện nay chủ yếu hỏi: *student nên bắt chước teacher ở mức output như thế nào?*

Ví dụ tiêu biểu

GKD – *On-Policy Distillation of Language Models: Learning from Self-Generated Mistakes* [3]
Distilling Step-by-Step! Outperforming Larger Language Models with Less Training Data and Smaller Model Sizes – Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alexander Ratner, Ranjay Krishna, Chen-Yu Lee, Tomas Pfister [8]
Direct Preference Optimization: Your Language Model is Secretly a Reward Model – Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, Chelsea Finn [9]

Topic 2: từ KL chuẩn sang objective thích nghi hơn

- Logit KD không còn dừng ở **forward KL**.
- Xu hướng rõ rệt là chuyển sang các objective: interpolation giữa forward / reverse KL, skew / adaptive divergence, contrastive / asymmetric objectives.
- Mục tiêu là xử lý tốt hơn chênh lệch năng lực giữa teacher và student.

Ví dụ tiêu biểu

On-Policy Distillation of Language Models: Learning from Self-Generated Mistakes – Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos, Matthieu Geist, Olivier Bachem [3]
DistiLLM: Towards Streamlined Distillation for Large Language Models – Jongwoo Ko, Sungnyun Kim, Tianyi Chen, Se-Young Yun [5]
DistiLLM-2: A Contrastive Approach Boosts the Distillation of LLMs – Jongwoo Ko, Tianyi Chen, Sungnyun Kim, Tianyu Ding, Luming Liang, Ilya Zharkov, Se-Young Yun [6]

Topic 3: từ static prefixes sang student-aware contexts

- Một hướng phát triển lớn là giảm **train–inference mismatch**.
- Thay vì luôn huấn luyện trên prefix cố định, student dần được học trên:
  - chính chuỗi do nó tự sinh,
  - hoặc các ngữ cảnh trộn teacher–student.
- Vì vậy, on-policy, mixed-context, và adaptive switching đang trở thành một trục phát triển quan trọng.

Ví dụ tiêu biểu

On-Policy Distillation of Language Models: Learning from Self-Generated Mistakes – Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos, Matthieu Geist, Olivier Bachem [3]
Speculative Knowledge Distillation: Bridging the Teacher-Student Gap Through Interleaved Sampling – Wenda Xu, Rujun Han, Zifeng Wang, Long T. Le, Dhruv Madeka, Lei Li, William Yang Wang, Rishabh Agarwal, Chen-Yu Lee, Tomas Pfister [4]
DistiLLM: Towards Streamlined Distillation for Large Language Models – Jongwoo Ko, Sungnyun Kim, Tianyi Chen, Se-Young Yun [5]

Topic 4: sequence distillation trở nên reasoning-aware

- Sequence distillation không còn chỉ học **đáp án cuối cùng**.
- Xu hướng mới là distill cả **quá trình suy luận**, như rationale, chain-of-thought, hoặc giải thích theo từng bước.
- Điều này cho thấy field đang chuyển từ

$$\text{imitate outputs} \rightarrow \text{imitate reasoning process.}$$

Ví dụ tiêu biểu

Distilling Step-by-Step! Outperforming Larger Language Models with Less Training Data and Smaller Model Sizes – Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alexander Ratner, Ranjay Krishna, Chen-Yu Lee, Tomas Pfister [8]

Speculative Knowledge Distillation: Bridging the Teacher-Student Gap Through Interleaved Sampling – Wenda Xu, Rujun Han, Zifeng Wang, Long T. Le, Dhruv Madeka, Lei Li, William Yang Wang, Rishabh Agarwal, Chen-Yu Lee, Tomas Pfister [4]

Topic 5: preference distillation đơn giản hóa alignment pipeline

- Preference distillation đang dịch chuyển từ pipeline kiểu **RLHF nhiều bước** sang các objective **trực tiếp và gọn hơn**.
- Điểm nhấn là bỏ reward model trung gian, tối ưu trực tiếp từ preference pairs, và tận dụng AI feedback / black-box teacher hiệu quả hơn.

Ví dụ tiêu biểu

Direct Preference Optimization: Your Language Model is Secretly a Reward Model – Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, Chelsea Finn [9]
Zephyr: Direct Distillation of LM Alignment – Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M. Rush, Thomas Wolf [10]

Topic 6: structural compatibility trở thành frontier thực dụng

- Community bắt đầu xem **compatibility giữa teacher và student** là vấn đề trung tâm, không chỉ là chi tiết kỹ thuật.
- Đặc biệt, **cross-tokenizer distillation** nổi lên như một frontier riêng: cần bridge giữa hai output spaces trước khi mới có thể distill.
- Đây là xu hướng rất thực tế vì nhiều teacher–student pairs hiện nay không còn dùng cùng tokenizer.

Ví dụ tiêu biểu

Dual-Space Knowledge Distillation for Large Language Models – Songming Zhang, Xue Zhang, Zengkui Sun, Yufeng Chen, Jinan Xu [11]

SRA: Span Representation Alignment for Large Language Model Distillation – Quoc Phong Dao, Hoang Son Nguyen, Pham Khanh Chi, Tung Nguyen, Linh Ngo Van, Diep Thi-Ngoc Nguyen, Trung Le [14]

Topic 7: Xu hướng phụ nhưng đáng chú ý: supervision vượt ra ngoài logits cuối

- Dù feature-based methods không phải dòng chính trong LLM KD, gần đây đã có xu hướng quay lại khai thác **động học biểu diễn theo layer**.
- Ý tưởng không chỉ là match output cuối, mà còn match **trajectory** và **delta** của feature qua các tầng.

Ví dụ tiêu biểu

Beyond Logits: Aligning Feature Dynamics for Effective Knowledge Distillation – Guoqiang Gong, Jiaxing Wang, Jin Xu, Deping Xiang, Zicheng Zhang, Leqi Shen, Yifeng Zhang, Junhua Shu, Zhaolong Xing, Zhen Chen, Pengzhang Liu, Ke Zhang [7] **MTA: Multi-Granular Trajectory Alignment for Large Language Model Distillation** – Pham Khanh Chi, Quoc Phong Dao, Thuat Nguyen, Linh Ngo Van, Trung Le, Thanh Nguyen [15]

On-Policy Distillation of Language Models: Learning from Self-Generated Mistakes

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos, Matthieu Geist, Olivier Bachem

[3]

2. On-policy Knowledge Distillation

On-policy KD

- Student tự sinh output sequence $y \sim p_S^\theta(\cdot \mid x)$.
- Teacher cho feedback token-level trên chuỗi đó (logits / probabilities), giúp giảm mismatch.
- Không backprop qua quá trình sampling của student $\Rightarrow$ training ổn định và rẻ hơn.

Generalized KD (GKD)

- GKD trộn hai loại dữ liệu:
  - dữ liệu cố định (X, Y),
  - dữ liệu on-policy do student sinh ra.
- Hệ số $\lambda \in [0, 1]$ điều khiển tỷ lệ dữ liệu on-policy trong loss tổng.

Loss on-policy KD và GKD

Loss on-policy KD

- Với input x và student policy $p_S^\theta(y \mid x)$:

$$\mathcal{L}_{OD}(\theta) = \mathbb{E}_{x\sim X}\left[\mathbb{E}_{y\sim p_S^\theta(\cdot|x)}\left[D_{KL}\left(p_T(\cdot \mid y_{<n}, x) \parallel p_S^\theta(\cdot \mid y_{<n}, x)\right)\right]\right].$$

Loss Generalized KD (GKD)

$$\mathcal{L}_{GKD}(\theta) = (1-\lambda)\,\mathbb{E}_{(x,y)\sim(X,Y)}\left[D\left(p_T \parallel p_S^\theta\right)(y \mid x)\right] + \lambda\,\mathbb{E}_{x\sim X}\left[\mathbb{E}_{y\sim p_S^\theta(\cdot|x)}\left[D\left(p_T \parallel p_S^\theta\right)(y \mid x)\right]\right].$$

Algorithm 1: Generalized KD (GKD)

Algorithm 1 Generalized Knowledge Distillation

Given: Teacher p_T, Student p_S^θ, dataset (X, Y)
Hyperparams: student fraction $\lambda \in [0, 1]$, divergence $\mathcal{D}$, lr η
for $k = 1, \ldots, K$ **do**
    Sample $u \sim \text{Uniform}(0, 1)$
    if $u \leq \lambda$ **then**
        Lấy x từ X, sinh $y \sim p_S^\theta(\cdot \mid x)$, được batch B
    else
        Lấy batch (x, y) từ (X, Y), được batch B
    end if
    Cập nhật θ:

$$\theta \leftarrow \theta - \eta \frac{1}{|B|} \sum_{(x,y) \in B} \nabla_\theta \mathcal{D}\big(p_T \parallel p_S^\theta\big)(y \mid x)$$

end for

Kết hợp RL Fine-tuning và On-policy GKD

Ý tưởng:

- Distillation chỉ cung cấp một *proxy* cho mục tiêu thực sự.
- Nếu mục tiêu chính là một reward không khả vi, ta có thể kết hợp trực tiếp RL với on-policy GKD.

Objective kết hợp RL + GKD:

$$\mathbb{E}_{x\sim X}\left[(1-\alpha)\,\mathbb{E}_{y\sim p_S^\theta(\cdot|x)}[r(y)] \;-\; \alpha\,\mathbb{E}_{y\sim p_S^\theta(\cdot|x)}\big[D(p_T \,\|\, p_S^\theta)(y \mid x)\big]\right]$$

- $\alpha = 1$: chỉ distillation.
- $\alpha < 1$: tối ưu reward nhưng vẫn giữ policy gần teacher.

Speculative Knowledge Distillation: Bridging the Teacher-Student Gap Through Interleaved Sampling

Wenda Xu, Rujun Han, Zifeng Wang, Long T. Le, Dhruv Madeka,
Lei Li, William Yang Wang, Rishabh Agarwal, Chen-Yu Lee, Tomas Pfister

[4]

3. Speculative Knowledge Distillation

- On-Policy KD giúp giảm mismatch nhưng có thể tạo ra mẫu chất lượng thấp, OOD đối với teacher.
- Supervised KD thì chất lượng training tốt hơn nhưng student không học cách sửa lỗi của chính nó.
- SKD được thiết kế để cân bằng giữa hai cực này thông qua **interleaved sampling**.

Speculative Knowledge Distillation (SKD)

Ý tưởng chính: Interleaved Sampling

- **Student** M_s đề xuất token ứng viên theo on-policy:

$$y_i \sim M_s(\cdot \mid y_{<i}, x_j).$$

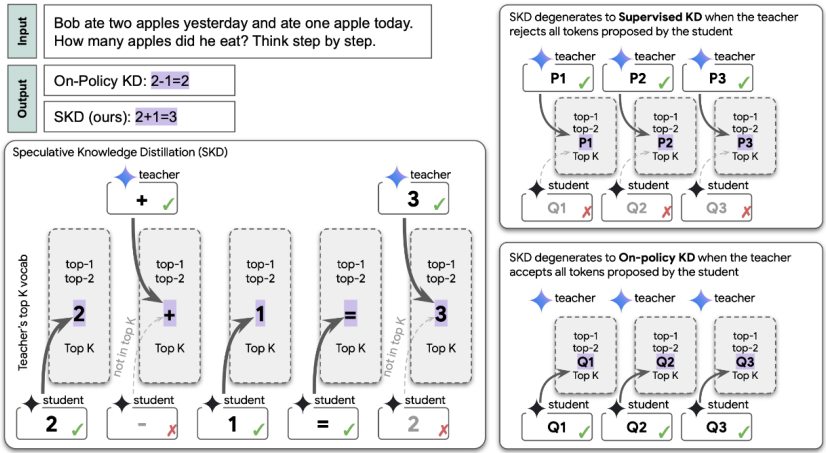
- **Teacher** M_t đánh giá song song các token này và chỉ chấp nhận token nằm trong top-K của M_t; các token còn lại được thay bằng token do teacher resample.

Cập nhật student

- Sau bước speculative sampling, ta có chuỗi interleaved (student + teacher).
- Tính lại phân phối teacher và student trên chuỗi này, rồi tối ưu loss

$$D(M_t \parallel M_s)(y \mid x).$$

SKD: minh họa


Input
Bob ate two apples yesterday and ate one apple today.
How many apples did he eat? Think step by step.
Output
On-Policy KD: 2-1=2
SKD (ours): 2+1=3
Speculative Knowledge Distillation (SKD)
Teacher's top K vocab
teacher
student
top-1
top-2
Top K
not in top K
SKD degenerates to Supervised KD when the teacher rejects all tokens proposed by the student
SKD degenerates to On-policy KD when the teacher accepts all tokens proposed by the student

DistiLLM: Towards Streamlined Distillation for Large Language Models

Jongwoo Ko, Sungnyun Kim, Tianyi Chen, Se-Young Yun

[5]

4. DistiLLM

Motivation

- KD cho LLM gặp 2 vấn đề lớn:
  1. KLD bất đối xứng $\Rightarrow$ gradient explosion / optimization instability.
  2. SGO-based KD có các hạn chế về nhiễu và chi phí.
- DistiLLM đề xuất:
  - **Skew-KLD** (ổn định tối ưu),
  - **Adaptive Off-Policy SGO** (hiệu quả hơn về tốc độ và sample efficiency).

Method: Skew KLD

Definitions

$$D_{\mathrm{SKL}}^{(\alpha)}(p,q) = D_{\mathrm{KL}}\big(p \,\big\|\, \alpha p + (1-\alpha)q\big), \qquad D_{\mathrm{SRKL}}^{(\alpha)}(p,q) = D_{\mathrm{KL}}\big(q \,\big\|\, (1-\alpha)p + \alpha q\big).$$

Stability key: Nếu $q(y|x) \to 0$ khi $p(y|x) > 0$,

$$r_{p,m_\alpha}(y|x) = \frac{p}{\alpha p + (1-\alpha)q} \;\le\; \frac{1}{\alpha},$$

nên trộn hai phân phối giúp tránh hiện tượng gradient explosion của KLD thông thường.

Adaptive SGO Scheduler

Ý tưởng chính:

- Dùng mẫu on-policy (SGO) với xác suất ϕ, dữ liệu cố định với xác suất $1 - \phi$.
- Không giữ ϕ cao cố định; scheduler bắt đầu thấp rồi tăng dần.

Lý do:

- ϕ thấp ban đầu giúp student không bị ngập trong nhiễu do SGO.
- Khi mô hình ổn định hơn, tăng ϕ để giảm train–inference mismatch.

Off-policy SGO for Sample Efficiency

Mục tiêu: tăng sample efficiency bằng chiến lược off-policy dựa trên replay buffer.

- Replay buffer D_R lưu các mẫu SGO từ student.
- Khi huấn luyện, rút mẫu ngẫu nhiên từ buffer để training.
- Xác suất dùng mẫu cũ giảm dần theo:

$$\lambda_R := \phi\left(1 - \frac{t}{T}\right).$$

DistiLLM-2: A Contrastive Approach Boosts the Distillation of LLMs

Jongwoo Ko, Tianyi Chen, Sungnyun Kim, Tianyu Ding,
Luming Liang, Ilya Zharkov, Se-Young Yun

[6]

5. DistiLLM-2

Từ góc độ loss

- KL divergence không nắm bắt đủ hành vi sinh mẫu phức tạp của teacher.
- Nhiều loss thay thế được đề xuất: Skew KL, contrastive hoặc asymmetric loss.

Từ góc độ dữ liệu

- Chỉ dựa vào TGOs gây mismatch training–inference.
- Một số hướng dùng SGOs để giảm mismatch nhưng chưa khai thác đủ sự cộng hưởng giữa loss và kiểu dữ liệu.

Phương pháp: Tiếp cận tương phản

- **Ý tưởng của DPO** [9]: tối đa hóa khoảng cách giữa preferred responses và dispreferred responses.
- **DPO áp dụng trực tiếp cho KD (DPKD)** thay mô hình tham chiếu bằng teacher p, nhưng dễ phạt quá mạnh SGOs.

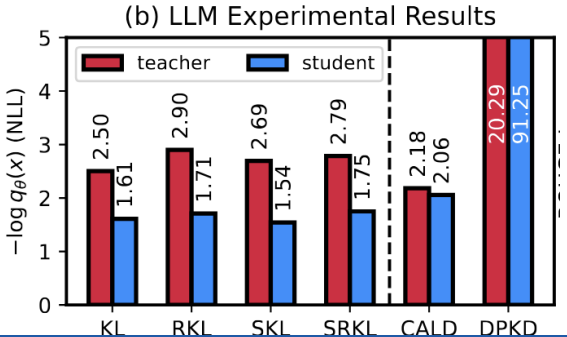
$$-\log \sigma\left(\lambda\left[\log \frac{q_\theta(y_t|x)}{p(y_t|x)} - \log \frac{q_\theta(y_s|x)}{p(y_s|x)}\right]\right)$$

Vấn đề của DPKD trên SGO

Đối với các SGO, teacher gần như không gán xác suất:

$$p(y_s \mid x) \approx 0 \Rightarrow \log \frac{q_\theta(y_s|x)}{p(y_s|x)} \text{ rất lớn}$$

- Phạt quá mạnh $q_\theta(y_s \mid x) \Rightarrow$ degenerate outputs / reward hacking.
- Dễ làm mô hình quên kiến thức tiền huấn luyện hữu ích.



Contrastive Approach for LLM Distillation

Giải pháp đề xuất: xây dựng hàm loss mới $\mathcal{L}_{\text{CALD}}$, kết hợp giữa:

- **SKL** cho phản hồi từ teacher (y_t),
- **SRKL** cho phản hồi từ student (y_s).

$$\mathcal{L}_{\text{CALD}} = (1-\beta)\,\text{SKL}\big(p \parallel q_\theta; \alpha_t\big) + \beta\,\text{SRKL}\big(q_\theta \parallel p; \alpha_s\big)$$

- SKL trên TGOs: kéo xác suất student lên ở vùng teacher mạnh.
- SRKL trên SGOs: đẩy xác suất student xuống ở tail của teacher.

Curriculum cho α và β

Mục tiêu:

- Curriculum cho hệ số α trong SKL/SRKL.
- Tăng dần hệ số của SRKL (β).

Trực giác:

- Mẫu dễ $\Rightarrow$ dùng α nhỏ.
- Mẫu khó $\Rightarrow$ dùng α lớn.
- Giai đoạn đầu tập trung học teacher behavior (SKL), giai đoạn sau tăng phản hồi từ SGOs (SRKL).

Beyond Logits: Aligning Feature Dynamics for Effective Knowledge Distillation

Guoqiang Gong, Jiaxing Wang, Jin Xu, Deping Xiang, Zicheng Zhang, Leqi Shen, Yifeng Zhang, Junhua Shu, Zhaolong Xing, Zhen Chen, Pengzhang Liu, Ke Zhang

[7]

6. Aligning Feature Dynamics for Effective Knowledge Distillation

Rời rạc hoá theo layer để khớp quỹ đạo giữa teacher và student:

$$\mathcal{L}_{\mathrm{KD}}^{\mathrm{Traj}} = \frac{1}{LN} \sum_{j=1}^{L_D} \sum_{i=1}^{N} D_{\mathrm{KL}}\Big(e^{y_i^T(l_j^T)} \,\Big\|\, e^{y_i^S(l_j^S)} \Big). \tag{1}$$

FDD: Discrete Layer-delta Derivative Loss

$$\mathcal{L}_{\text{KD}}^{\text{Der}} = \frac{1}{NL_{\Delta}} \sum_{j=1}^{L_{\Delta}} \sum_{i=1}^{N} D_{\text{Cos}}\Big(\Delta_i^T(I_{l_j}^T) \,\Big\|\, \Delta_i^S(I_{l_j}^S)\Big), \tag{2}$$

với

$$\Delta_i^T(I_{l_j}^T) = y_i^T(I_{l_j}^T) - y_i^T(I_{l_{j-1}}^T), \tag{3}$$

$$\Delta_i^S(I_{l_j}^S) = y_i^S(I_{l_j}^S) - y_i^S(I_{l_{j-1}}^S). \tag{4}$$

Overall FDD Objective

$$\mathcal{L}_{\text{overall}} = \mathcal{L}_{\text{KD}} + \alpha \mathcal{L}_{\text{KD}}^{\text{Traj}} + \beta \mathcal{L}_{\text{KD}}^{\text{Der}}.$$

- $\mathcal{L}_{\text{KD}}$: loss trên logits cuối.
- $\mathcal{L}_{\text{KD}}^{\text{Traj}}$: loss cho quỹ đạo layer-wise.
- $\mathcal{L}_{\text{KD}}^{\text{Der}}$: loss cho đạo hàm / layer-delta.

MTA: Multi-Granular Trajectory Alignment for Large Language Model Distillation

Pham Khanh Chi, Quoc Phong Dao, Thuat Nguyen,
Linh Ngo Van, Trung Le, Thanh Nguyen

[15]

7. MTA: động lực

Hạn chế của các phương pháp hiện tại

- Đa số align tại **layer cố định** hoặc chỉ ở mức token output, bỏ qua cách biểu diễn *tiến hóa theo độ sâu*.
- FDD đã distill feature dynamics nhưng dùng **cùng một objective** cho mọi layer được chọn, không mã hóa cấu trúc ngôn ngữ phụ thuộc độ sâu.

MTA: động lực

Quan sát nền tảng

- Ngôn ngữ có cấu trúc **phân cấp**: từ vựng hợp thành cụm từ, cụm từ hợp thành ngữ nghĩa cấp cao.
- Nghiên cứu interpretability: tầng thấp mã hóa đặc trưng bề mặt và từ vựng, tầng cao mã hóa ngữ nghĩa trừu tượng.

Kết luận: biểu diễn tạo thành một **hierarchical representational trajectory**, nên áp một luật align đồng nhất cho mọi layer là dưới tối ưu.

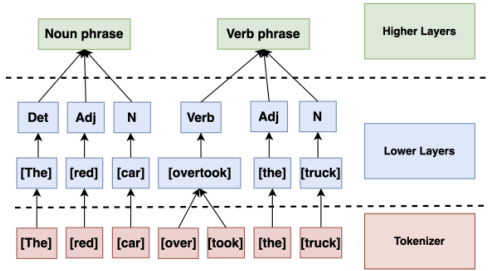
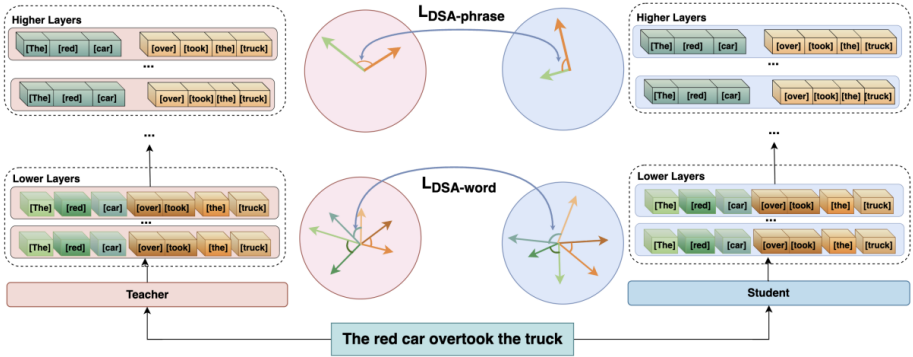


Figure 1: The correspondence between linguistic compositionality and the layer-wise evolution of representations in large language models.

Chiến lược phân tầng

- **Tầng thấp**: gom token thành **Word Spans**, tức gộp các sub-word của cùng một từ, để giữ nền tảng từ vựng.
- **Tầng cao**: dùng **Phrase Spans** là Noun Phrase và Verb Phrase, trích bằng spaCy, để nắm ngữ nghĩa tổ hợp.



Token Weight và Span Representation

Vấn đề: attention gốc của LLM là causal nên thiên vị token đứng sớm, vì chúng đơn giản là được nhiều vị trí sau nhìn thấy hơn.

Giải pháp: tính lại trọng số bằng self-attention hai chiều, có mask đường chéo để loại self-loop.

1. Chuẩn hóa hidden state tại tầng l: $\hat{H}_{t,l} = H_{t,l} \,/\, \sigma(H_{t,l})$
2. Tính điểm tương đồng theo cặp, kèm mask:
$$S_{s\to t,l} = \frac{\hat{H}_{s,l}\hat{H}_{t,l}^{\top}}{\sqrt{d}} + M_{s,t}, \qquad M_{s,t} = -\infty \;\text{ nếu } s = t \text{ hoặc } t \in \mathcal{P}$$
3. Softmax rồi lấy trung bình attention mà token t *nhận được*:
$$w_{t,l} = \frac{1}{N}\sum_{s=1}^{N} \alpha_{s\to t,l}, \qquad \alpha_{s\to t,l} = \mathrm{softmax}_t\big(S_{s\to t,l}\big)$$
4. Gộp token thành span bằng trung bình có trọng số:
$$U_{k,l} = \frac{\sum_{t\in S_k} w_{t,l} H_{t,l}}{\sum_{t\in S_k} w_{t,l}}$$

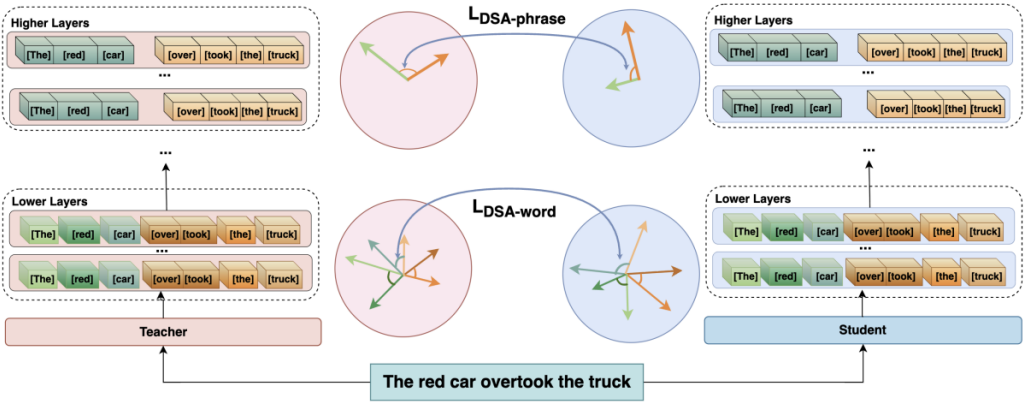
Dynamic Structural Alignment Loss

$$\mathcal{L}_{DSA} = \frac{1}{\|L_{key}\|} \sum_{l \in L_{key}} \mathcal{L}_{DSA}^{(l)}$$

$$\mathcal{L}_{DSA}^{(l)} = \sum_{i=1}^{N_l^{sp}} \sum_{j=i+1}^{N_l^{sp}} w_{ij,\phi(l)}^{sp} \Big(d(U_{i,l}^S, U_{j,l}^S) - d(U_{i,\phi(l)}^T, U_{j,\phi(l)}^T) \Big)^2$$

- $d(\cdot,\cdot)$ là cosine distance, $\phi(\cdot)$ là hàm ánh xạ layer student sang layer teacher.
- Trọng số cặp span $w_{ij,\phi(l)}^{sp} = w_{i,\phi(l)}^{sp} w_{j,\phi(l)}^{sp}$ ưu tiên các span nổi bật về ngữ nghĩa.
- Loss này ràng buộc **quan hệ hình học trong từng layer** và cách quan hệ đó biến đổi theo độ sâu.

Dynamic Structural Alignment Loss: Minh hoạ



Hình: Dynamic Structural Alignment ($\mathcal{L}_{DSA}$). Với mỗi tầng được chọn, ta tính ma trận khoảng cách cosine giữa các span của teacher và của student, rồi buộc hai cấu trúc quan hệ này trùng nhau. Tầng thấp dùng word span, tầng cao dùng phrase span, nhờ đó student học lại được quỹ đạo biểu diễn của teacher theo độ sâu.

Hidden Representation Alignment và Overall Loss

Chiếu student sang không gian teacher: $\tilde{H}^S_{t,l} = H^S_{t,l} W_l$, sau đó

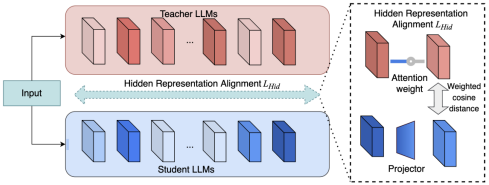
$$\mathcal{L}_{Hid} = \sum_{l \in L_{key}} \sum_{t \in \mathcal{M}_l} w^T_{t,l} \left(1 - \frac{\langle \tilde{H}^S_{t,l}, H^T_{t,l} \rangle}{\|\tilde{H}^S_{t,l}\|_2 \|H^T_{t,l}\|_2} \right)$$

với $\mathcal{M}_l$ là tập token nằm trong các span được trích tại tầng l, và $w^T_{t,l}$ là trọng số token lấy từ teacher.

MTA là một plug-in module:

$$\mathcal{L}_{Total} = \mathcal{L}_{Base} + \lambda_{DSA} \mathcal{L}_{DSA} + \lambda_{Hid} \mathcal{L}_{Hid}$$

trong đó $\mathcal{L}_{Base}$ có thể là DistiLLM, DistiLLM-2 hoặc FDD.



Student khớp biểu diễn của teacher qua một projector tuyến tính, dùng cosine distance có trọng số theo độ quan trọng của token tại các tầng được chọn.

Explain in Your Own Words: Improving Reasoning via Token-Selective Dual Knowledge Distillation

Minsang Kim, Seung Jun Baek

[12]

8. TSD-KD

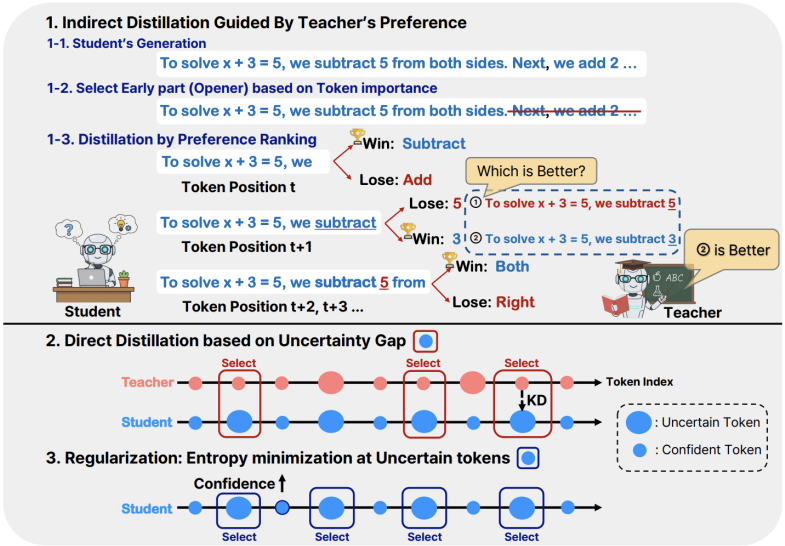
Motivation: Giảm Model Capacity Mismatch giữa teacher và student

- **Entropy:** Các token có entropy cao thường đóng vai trò là các điểm rẽ nhánh quan trọng trong quá trình suy luận.
- **Vị trí:** Những token có entropy cao thường tập trung xuất hiện ở phần đầu của response.

Phương pháp đề xuất: Chỉ cung cấp hướng dẫn ở các token quan trọng. Gồm 3 thành phần loss:

- **Indirect Distillation:** Học xếp hạng logit của teacher trên các token ở đầu chuỗi reasoning.
- **Direct Distillation:** Tập trung vào các token mà teacher chắc chắn và student không chắc chắn.
- **Entropy Regularization:** Tăng sự tự tin (confidence) cho student ở các token khó.

TSD: Minh họa


1. Indirect Distillation Guided By Teacher's Preference
1-1. Student's Generation
To solve x + 3 = 5, we subtract 5 from both sides. Next, we add 2 ...
1-2. Select Early part (Opener) based on Token importance
To solve x + 3 = 5, we subtract 5 from both sides. Next, we add 2 ...
1-3. Distillation by Preference Ranking
To solve x + 3 = 5, we
Token Position t
Win: Subtract
Lose: Add
To solve x + 3 = 5, we subtract
Token Position t+1
Lose: 5
Win: 3
To solve x + 3 = 5, we subtract 5 from
Token Position t+2, t+3 ...
Win: Both
Lose: Right
Which is Better?
① To solve x + 3 = 5, we subtract 5
② To solve x + 3 = 5, we subtract 3
② is Better
Student
Teacher
2. Direct Distillation based on Uncertainty Gap
Select
Select
Select
Teacher
Student
KD
Token Index
: Uncertain Token
: Confident Token
3. Regularization: Entropy minimization at Uncertain tokens
Confidence
Student
Select
Select
Select
Select

Indirect Distillation

Thay vì bắt student sao chép hoàn toàn xác suất của teacher:

- Với mỗi token, student đưa ra top k token dự đoán có logit cao nhất.
- Teacher đóng vai trò như một bộ định hướng, xếp hạng (re-rank) các token này, student học lại xếp hạng này bằng phân phối Plackett-Luce.
- Chỉ áp dụng giới hạn ở phần đầu của chuỗi suy luận (dựa trên phần trăm entropy tích lũy, tổng entroy $<$ thresshold c) để định hướng tư duy ban đầu.

$$\mathcal{L}_{\text{Indirect}} = -\sum_{t} \log P_{PL}(\pi_t \mid x, y_{<t})$$

Direct Distillation

Giải quyết các trường hợp student không tự tin ở 1 số token:

- Sử dụng hàm cổng (gating function) σ_τ dựa trên hiệu số entropy giữa student và teacher: $H_t(p_S) - H_t(p_T)$.
- Mô hình chỉ tập trung chưng cất mạnh ở những token mà student rất bối rối (uncertain) nhưng teacher lại cực kỳ chắc chắn (confident).

$$\mathcal{L}_{\text{Direct}} = \frac{1}{L} \sum_{t=1}^{L} \sigma_\tau \Big(H_t(p_S) - H_t(p_T) \Big) \cdot D_{\text{JSD}} \big(p_T \parallel p_\theta^S \big)$$

Entropy Regularization

Mục tiêu:

- Giảm sự không chắc chắn ở các bước lý luận trung gian.
- Duy trì và gia tăng sự tự tin (confidence) cho student trong quá trình chưng cất tri thức, đặc biệt ở các điểm rẽ nhánh quan trọng.

Entropy Regularization:

$$\mathcal{L}_{\text{EM}} = \mathbb{E}_{x \sim X} \left[\frac{1}{|\mathcal{I}|} \sum_{t \in \mathcal{I}} H_t(p_\theta^S) \right]$$

- *Trong đó:* $H_t(p_\theta^S)$ là entropy của student tại bước t và $|\mathcal{I}|$ là số lượng token thuộc top 10% có entropy.

Overall Loss

$$\min_{\theta} \mathbb{E}_{x \sim X} \left[\alpha \mathcal{L}_{\text{Indirect}} + \mathcal{L}_{\text{Direct}} + \mathcal{L}_{\text{EM}} \right]$$

- $\mathcal{L}_{\text{Indirect}}$: Indirect Distillation Loss.
- $\mathcal{L}_{\text{Direct}}$: Direct Distillation Loss.
- $\mathcal{L}_{\text{EM}}$: Entropy Regularization.
- Hệ số $\alpha > 0$ được sử dụng để kiểm soát mức độ tác động của chưng cất gián tiếp

Distillation of Large Language Models via Concrete Score Matching

Yeongmin Kim, Donghyeok Shin, Mina Kang, Byeonghu Na, Il-Chul Moon

[13]

Softmax smoothing

- Hầu hết các phương pháp KD dùng KL divergence để khớp phân phối xác suất qua hàm Softmax.
- Softmax làm mất đi thông tin giá trị logit cực kỳ quan trọng của teacher (thường gán xác suất gần bằng 0 cho phần lớn token trong từ vựng).

Hạn chế của Direct Logit Distillation (DLD)

- Khớp trực tiếp logit (DLD) giúp giữ lại thông tin, nhưng lại thất bại trong việc xử lý *tính bất biến dịch chuyển logit (logit shift invariance: softmax(logits) = softmax(logits + C))*.
- Việc ép buộc logit của student phải bằng chính xác logit của teacher làm thu hẹp nghiêm trọng không gian nghiệm tối ưu của mô hình.

Phương pháp: Concrete Score Distillation

Giải pháp đề xuất: Concrete Score Distillation (CSD)

- Khắc phục cả hai nhược điểm: loại bỏ softmax smoothing và mở rộng lại không gian nghiệm.
- Áp dụng ý tưởng từ mô hình dựa trên năng lượng (EBM) và score-matching cho biến rời rạc (concrete score).

Cơ chế hoạt động:

- CSD không đối chiếu các logit một cách trực tiếp, mà tiến hành căn chỉnh **chênh lệch logit tương đối (logit residuals)** giữa mọi cặp token trong từ vựng của student và teacher.
- Phương pháp này cho phép áp dụng các trọng số linh hoạt để mô hình hoá được mối quan hệ nội tại của các từ vựng.

3. CSD Objective

Với mỗi token:

$$\mathcal{L}_{\text{CSD}}(\theta; p_T, w) = \frac{1}{2} \sum_{y_t \in \mathcal{V}} \sum_{x \in \mathcal{V}} w(y_t, x) \Big(f_\theta[x] - f_\theta[y_t] - f_T[x] + f_T[y_t] \Big)^2$$

Trong đó:

- f_θ, f_T: Hàm tính logit của student và teacher tại các token x và y_t của mỗi token t.
- $w(y_t, x)$: Hàm trọng số để phân bổ độ quan trọng cho các cặp token.
- Student so sánh sự khác biệt nội tại giữa các token của chính nó và điều chỉnh sao cho giống với khoảng chênh lệch tương ứng của teacher.

Cơ chế tính toán $w(y_t, x)$

CSD tách hàm trọng số thành 2 thành phần độc lập:

$$w(y_t, x) = w_1(y_t) \times w_2(x)$$

Các thành phần w_1 và w_2 là các phân phối xác suất trên toàn bộ từ vựng $\mathcal{V}$:

- **Phân phối Học sinh (S - Student):** Lấy trực tiếp xác suất sinh từ của mô hình học sinh. Được ngắt gradient
- **Phân phối Giáo viên (T - Teacher):** Lấy xác suất sinh từ của giáo viên
- **Phân phối Đồng đều (U - Uniform):** Gán trọng số bằng nhau cho mọi token.

$$w(x) = \frac{1}{|\mathcal{V}|}$$

Cấu hình CSD

- **CSD (S, S):** $w(y_t, x) = q_\theta(y_t) \times q_\theta(x)$. Chỉ tập trung học ở vùng học sinh đã quen thuộc và tự tin $\rightarrow$ Tối đa hóa **độ trung thực (fidelity)**.
- **CSD (U, S):** $w(y_t, x) = \frac{1}{|\mathcal{V}|} \times q_\theta(x)$. Ép học sinh phải học dàn đều cả các từ hiếm $\rightarrow$ Tăng cường cực mạnh **tính đa dạng (diversity)**.
- **CSD (T, S):** $w(y_t, x) = p_T(y_t) \times q_\theta(x)$. Dùng xác suất của Giáo viên làm mỏ neo định hướng $\rightarrow$ Đạt **độ hiệu chuẩn xác suất (calibration)** tốt nhất.

SRA: Span Representation Alignment for Large Language Model Distillation

Quoc Phong Dao, Hoang Son Nguyen, Pham Khanh Chi, Tung Nguyen, Linh Ngo Van, Diep Thi-Ngoc Nguyen, Trung Le

[14]

10. Motivation: SRA

Vấn đề: Cross-Tokenizer KD (CTKD)

- KD cổ điển giả định teacher và student dùng **cùng tokenizer**. Giả định này thường bị vi phạm trong thực tế.
- Các hướng hiện có: align chuỗi token bằng edit distance, DTW, hoặc chuyển toàn bộ phân phối bằng Optimal Transport.
- Điểm yếu chung: vẫn thao tác trên **đơn vị token rời rạc**, vốn rất nhạy cảm với khác biệt tokenizer.

Ý tưởng của SRA

- Đổi đơn vị align từ **token** sang **span** (đoạn văn bản), vốn bất biến với tokenizer.
- Nhìn Transformer như **Multi-Particle Dynamical System**: token là các hạt, hidden state là vị trí, layer là bước thời gian.
- Mỗi span là một **cụm hạt**, được biểu diễn bằng **tâm khối (Center of Mass)**.

Background: Transformer như Multi-Particle Dynamical System

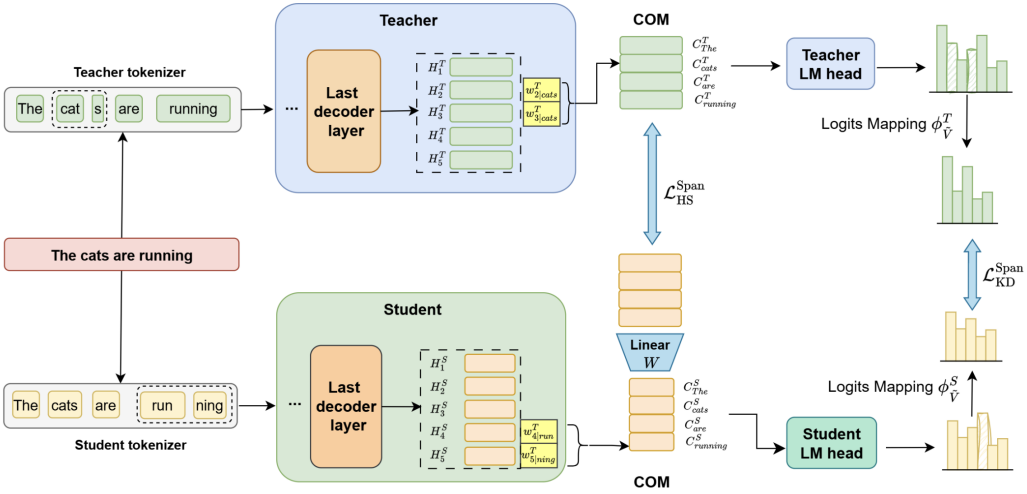
Transformer	**Hệ nhiều hạt**
Hidden state $h_{\ell,i}$	Vị trí hạt $x_{i,t} \in \mathbb{R}^d$
Chỉ số layer ℓ	Bước thời gian rời rạc t
Multi-Head Self-Attention	Khuếch tán: tương tác giữa các hạt
Feed-Forward Network	Đối lưu: cập nhật riêng của từng hạt
Residual update	Bước tách Lie–Trotter

Tính chất then chốt của tâm khối: tâm khối của một cụm lớn có thể tính *phân cấp* từ tâm khối của các cụm con:

$$\tilde{x} = \frac{1}{M} \sum_{k=1}^{K} M_k \tilde{x}_k, \qquad \tilde{x}_k = \frac{1}{M_k} \sum_{i \in C_k} m_i x_i.$$

Vì vậy span representation nên là **tổ hợp tuyến tính** của các token thành phần.

Minh Hoạ: SRA


Teacher
Teacher tokenizer
The cat s are running
Last decoder layer
COM
Teacher LM head
Logits Mapping $\phi^T_{\hat{V}}$
The cats are running
Student
Linear W
Last decoder layer
Student tokenizer
The cats are run ning
Student LM head
Logits Mapping $\phi^S_{\hat{V}}$
COM
$\mathcal{L}^{\text{Span}}_{\text{HS}}$
$\mathcal{L}^{\text{Span}}_{\text{KD}}$

Span Mapping bằng LCS

Cùng một câu x được tokenize bởi hai tokenizer, sinh ra hai chuỗi **character offset**:

$$(\text{off}_1^T, \ldots, \text{off}_{n_T}^T),\ (\text{off}_1^S, \ldots, \text{off}_{n_S}^S).$$

- Tính **Longest Common Subsequence** của hai chuỗi offset để tìm các điểm cắt chung.
- Mỗi cặp điểm cắt liên tiếp xác định một cặp span khớp nhau:

$$M_s = [(\text{Span}_0^S, \text{Span}_0^T), \ldots].$$

- Token đặc biệt (ví dụ `[PAD]`) được gán offset 0 và bỏ qua.
- Span-level alignment **phủ toàn bộ câu**, hạn chế mất mát thông tin.

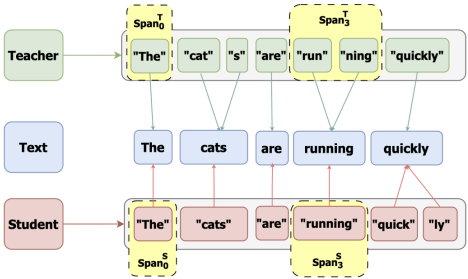


Figure 1: An illustration of the tokenizer mismatch between a teacher and a student model.

Span representation qua Weighted Token Pooling

Trọng số token (vai trò khối lượng hạt) tại layer cuối L:

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

với A_h là số head, $\mathrm{Att}_{a,N,i}$ là attention từ token cuối N tới token i.

Center of Mass của span:

$$C_i^S = \sum_{t=s_i^S}^{e_i^S} w_t^S H_t^S, \quad C_i^T = \sum_{t=s_i^T}^{e_i^T} w_t^T H_t^T.$$

Trong đó (s_i, e_i) là chỉ số token bắt đầu và kết thúc của span thứ i, H_t là hidden state ở tầng cuối của token t, w_t là trọng số token tương ứng, và các chỉ số trên S, T ký hiệu student và teacher.

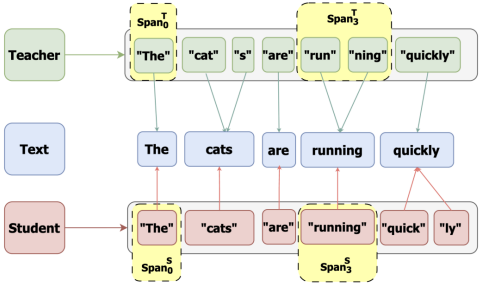


Figure 1: An illustration of the tokenizer mismatch between a teacher and a student model.

Span-Level Hidden State Transfer

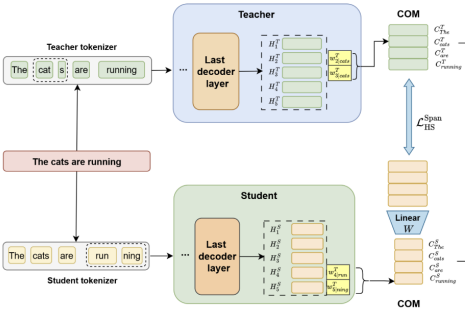
$$\mathcal{L}_{HS}^{Span} = \sum_{i=1}^{N_{sp}} w_i^{sp}\, \mathcal{L}_{\cos}\big(C_i^S W, C_i^T\big) + \lambda \mathcal{L}_{Geo}$$

với $\mathcal{L}_{\cos}(u, v) = 1 - \frac{u \cdot v}{\|u\|\|v\|}$, N_{sp} là số cặp span đã khớp, và W là ma trận chiếu học được.

Trọng số span tái sử dụng trọng số token của teacher:

$$w_i^{sp} = \frac{\tilde{w}_i^{sp}}{\sum_t \tilde{w}_t^{sp}}, \qquad \tilde{w}_i^{sp} = \Big(\sum_{t=s_i^T}^{e_i^T} w_t^T \Big)^p$$

Tham số p điều khiển độ sắc của phân phối trọng số; $p = 0$ nghĩa là mọi span quan trọng như nhau.



Span của student được chiếu qua W rồi khớp với span tương ứng của teacher, có trọng số theo độ nổi bật.

Geometric Regularizer

Vấn đề: phép chiếu tuyến tính không ràng buộc có thể làm méo hình học của không gian biểu diễn.

Giải pháp: giữ nguyên quan hệ tương đối giữa các span

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

với

$$w_{i,j}^{sc} = \frac{w_i^{sp} w_j^{sp}}{\sum_i \sum_{j>i} w_i^{sp} w_j^{sp}}, \qquad \sum_{i<j} w_{i,j}^{sc} = 1.$$

Chuẩn hóa giúp loss so sánh được giữa các mẫu có số span khác nhau.

Span-Level Logits Transfer

Góc nhìn: logits của một token là vị trí của hạt được chiếu vào không gian từ vựng, nên có thể gộp lên mức span.

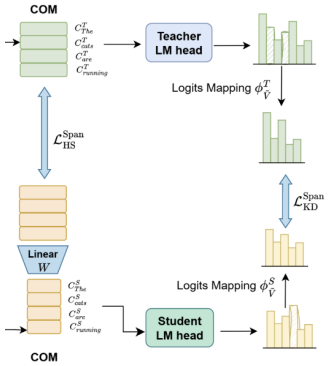
Vấn đề $V_{tea} \neq V_{stu}$ được xử lý bằng cách làm việc trên **không gian con chung** $\tilde{V} = V_{tea} \cap V_{stu}$:

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

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

$$\mathcal{L}^{Span}_{KD} = \sum_{i=1}^{N_{sp}} \mathcal{L}_{KL}\left(\tilde{y}^T_i, \tilde{y}^S_i; \tau\right)$$

f_{head} là LM head, $\phi_{\tilde{V}}$ là hàm chiếu logits vào từ vựng chung, τ là nhiệt độ softmax.



SRA: Overall objective

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

- $\mathcal{L}_{CE}$: cross-entropy trên nhãn thật, giữ tín hiệu giám sát của tác vụ.
- $\mathcal{L}_{HS}^{Span}$: khớp biểu diễn span ở tầng cuối, kèm ràng buộc hình học $\mathcal{L}_{Geo}$.
- $\mathcal{L}_{KD}^{Span}$: khớp phân phối logits mức span trên từ vựng chung $\tilde{V}$.
- $\alpha \in [0, 1]$ cân bằng giữa học từ nhãn và học từ teacher. Trong thí nghiệm, α nằm trong khoảng 0.5 đến 0.8 tùy cặp mô hình, student càng lớn thì α càng cao.

Siêu tham số chung: $\tau = 2.0$, $p = 1.0$, $\lambda = 50$.

TALAS: Teacher-Anchored Layer Alignment with Adaptive Sharpness-Aware Minimization for Embedding Distillation

Quoc Phong Dao, Hoang Son Nguyen, Pham Khanh Chi, Linh Ngo Van, Diep Thi-Ngoc Nguyen, Thien Huu Nguyen, Trung Le

[16]

11. TALAS: động lực

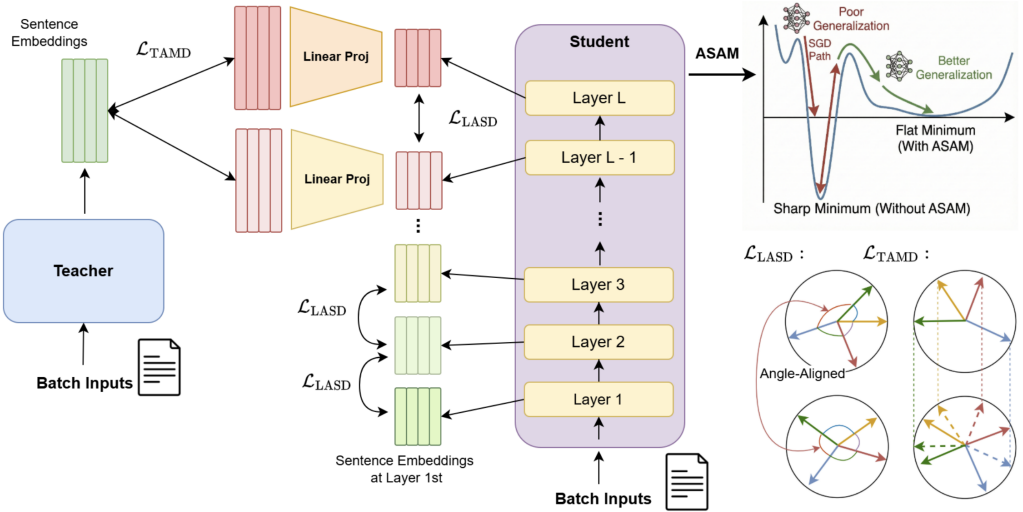
Bối cảnh: text embedding models là thành phần lõi của RAG, nhưng các model SOTA rất lớn.
Đánh đổi trong embedding distillation

- Khai thác biểu diễn nội bộ (hidden states, attention) cho tín hiệu giàu nhưng **tốn bộ nhớ và thời gian**, đặc biệt khi teacher là LLM-based.
- Chỉ dùng output embedding thì rẻ hơn nhiều nhưng **nén quá nhiều ngữ nghĩa vào một vector**, cộng với capacity gap sẽ khiến student bỏ sót sắc thái.

Ba thành phần của TALAS

- **Teacher-Anchored Multi-Layer Distillation**: chỉ neo các tầng trên vào embedding teacher.
- **Layer-Aligned Self-Distillation**: lan truyền cấu trúc quan hệ từ trên xuống dưới, không cần hidden state của teacher.
- **ASAM**: tối ưu hướng tới flat minima để chống overfit nhiễu điểm của teacher.

TALAS: minh họa



Teacher-Anchored Multi-Layer Distillation

Coi embedding cuối của teacher như một **mỏ neo tĩnh** giám sát nhiều tầng trên của student:

$$\mathcal{L}_{TAMD}(l) = \frac{1}{N}\sum_{i=1}^{N} \mathcal{D}_{\cos}\left(e_{l,i}^{S} W_l, e_i^{T}\right), \qquad \mathcal{L}_{TAMD} = \frac{1}{k+1}\sum_{l=L-k}^{L} \mathcal{L}_{TAMD}(l)$$

- $\mathcal{D}_{\cos}(u, v) = 1 - \frac{u^\top v}{\|u\|\|v\|}$, W_l là ma trận chiếu học được tại tầng l.
- Chỉ áp dụng cho **tầng trên**, nơi student đủ năng lực xấp xỉ không gian teacher.
- Tạo hiệu ứng *early semantic injection*: tầng trung gian nhận ngữ nghĩa cấp cao trước khi hợp nhất ở tầng cuối.
- Teacher chỉ chạy **một lần** lúc tiền xử lý, embedding được cache lại, nên training không cần teacher inference.

Layer-Aligned Self-Distillation

Vấn đề với tầng thấp: ép tầng nông khớp trực tiếp biểu diễn trừu tượng của teacher sẽ phá hỏng học đặc trưng, còn để tầng nông hoàn toàn tự do lại tạo đứt gãy biểu diễn theo độ sâu.

Giải pháp: dùng tầng $l+1$ của chính student làm "trợ giảng"cho tầng l, theo tinh thần Relational KD.

$$\mathrm{R}_l^S = \tilde{\mathrm{E}}_l^S (\tilde{\mathrm{E}}_l^S)^\top, \qquad \mathcal{L}_{LASD} = \frac{1}{L-1} \sum_{l=1}^{L-1} \frac{1}{N^2} \left\| \mathrm{R}_{l+1}^S - \mathrm{R}_l^S \right\|_F^2$$

- R_l^S là ma trận quan hệ cosine giữa các mẫu trong batch, mô tả **topology hình học** của không gian tại tầng l.
- Tri thức của teacher neo ở đỉnh được **chuyển dần xuống dưới** qua chuỗi ràng buộc cấu trúc.

Objective tổng và tối ưu bằng ASAM

Thành phần contrastive: tránh collapse và anisotropy kế thừa từ teacher

$$\mathcal{L}_{SimCSE} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{\mathrm{sim}(e_i^z, e_i^{z'})/\tau}}{\sum_{j=1}^{N} e^{\mathrm{sim}(e_i^z, e_j^{z'})/\tau}}$$

$$\mathcal{L}_{total} = \lambda_1\mathcal{L}_{SimCSE} + \lambda_2\mathcal{L}_{TAMD} + \lambda_3\mathcal{L}_{LASD}$$

ASAM: giải bài toán min-max cục bộ

$$\min_{w}\max_{\|\mathrm{T}_w^{-1}\epsilon\|_2\le\rho}\mathcal{L}(w+\epsilon)$$

với T_w chuẩn hóa nhiễu theo độ lớn tham số, giúp sharpness bất biến với scale. Cần hai lượt forward-backward mỗi bước.

Algorithm: TALAS Training via ASAM

Algorithm 2 TALAS Training via ASAM

Input: corpus $\mathcal{D}$, teacher f_T, student $f_S(\cdot; w)$, lr η, bán kính ρ
Tiền tính và cache embedding teacher: $\mathrm{D}_e^T \leftarrow \{f_T(x)_{anchored}\}_{x \in \mathcal{D}}$
for mỗi bước huấn luyện **do**
    Lấy mini-batch $\mathcal{B} \subset \mathcal{D}$
    Lấy target đã cache $\mathrm{E}^T \in \mathrm{D}_e^T$ cho $\mathcal{B}$
    Ascent step: $\mathcal{L} \leftarrow \mathcal{L}_{total}(f_S(\mathcal{B}; w), \mathrm{E}^T)$
    $\tilde{w} \leftarrow \mathrm{ASAM\_Perturb}(w, \nabla_w \mathcal{L}, \rho)$
    Descent step: $g_{ASAM} \leftarrow \nabla_{\tilde{w}} \mathcal{L}_{total}(f_S(\mathcal{B}; \tilde{w}), \mathrm{E}^T)$
    $w \leftarrow w - \eta \cdot g_{ASAM}$
end for

Conclusion

- **Từ nén mô hình đến chuyển giao năng lực:** KD cho LLM không chỉ nhằm giảm kích thước mô hình, mà hướng tới chuyển giao tri thức, hành vi và năng lực từ teacher sang student.
- **Từ teacher-forced đến student-aware:** On-policy và interleaved sampling khai thác chính các output do student sinh ra, qua đó giảm khoảng cách giữa quá trình huấn luyện và suy luận.
- **Từ bắt chước output đến giám sát đa dạng:** Reasoning traces, preference signals, feature dynamics và representation ngày càng được sử dụng bên cạnh logits đầu ra.
- **Từ token-level đến structure-aware:** Cross-tokenizer KD thúc đẩy việc căn chỉnh ở mức span và các đơn vị biểu diễn ổn định hơn thay vì phụ thuộc trực tiếp vào tokenizer.

Student-aware – Structure-aware – Capability-aware

Chuyển giao không chỉ "student nên dự đoán gì", mà còn "teacher biểu diễn và suy luận như thế nào" và "student có thể tiếp thu được gì".

References I

Hinton, G., Vinyals, O., & Dean, J. (2015).
Distilling the Knowledge in a Neural Network.
arXiv preprint arXiv:1503.02531.

Kim, Y., & Rush, A. M. (2016).
Sequence-Level Knowledge Distillation.
EMNLP 2016.

Agarwal, R., Vieillard, N., Zhou, Y., Stanczyk, P., Ramos, S., Geist, M., & Bachem, O. (2024).
On-Policy Distillation of Language Models: Learning from Self-Generated Mistakes.
ICLR 2024.

Xu, W., Han, R., Wang, Z., Le, L. T., Madeka, D., Li, L., Wang, W. Y., Agarwal, R., Lee, C.-Y., & Pfister, T. (2024).
Speculative Knowledge Distillation: Bridging the Teacher-Student Gap Through Interleaved Sampling.
arXiv preprint arXiv:2410.11325.

Ko, J., Kim, S., Chen, T., & Yun, S.-Y. (2024).
DistiLLM: Towards Streamlined Distillation for Large Language Models.
ICML 2024.

References II

Ko, J., Chen, T., Kim, S., Ding, T., Liang, L., Zharkov, I., & Yun, S.-Y. (2025).
DistiLLM-2: A Contrastive Approach Boosts the Distillation of LLMs.
arXiv preprint arXiv:2503.07067.

Gong, G., Wang, J., Xu, J., Xiang, D., Zhang, Z., Shen, L., Zhang, Y., Shu, J., Xing, Z., Chen, Z., Liu, P., & Zhang, K. (2025).
Beyond Logits: Aligning Feature Dynamics for Effective Knowledge Distillation.
ACL 2025.

Hsieh, C.-Y., Li, C.-L., Yeh, C.-K., Nakhost, H., Fujii, Y., Ratner, A., Krishna, R., Lee, C.-Y., & Pfister, T. (2023).
Distilling Step-by-Step! Outperforming Larger Language Models with Less Training Data and Smaller Model Sizes.
Findings of ACL 2023.

Rafailov, R., Sharma, A., Mitchell, E., Ermon, S., Manning, C. D., & Finn, C. (2023).
Direct Preference Optimization: Your Language Model is Secretly a Reward Model.
NeurIPS 2023.

Tunstall, L., Beeching, E., Lambert, N., Rajani, N., Rasul, K., Belkada, Y., Huang, S., von Werra, L., Fourrier, C., Habib, N., Sarrazin, N., Sanseviero, O., Rush, A. M., & Wolf, T. (2023).
Zephyr: Direct Distillation of LM Alignment.
arXiv preprint arXiv:2310.16944.

References III

Zhang, S., Zhang, X., Sun, Z., Chen, Y., & Xu, J. (2024).
Dual-Space Knowledge Distillation for Large Language Models.
EMNLP 2024.

Kim, M., & Baek, S. J. (2026).
Explain in Your Own Words: Improving Reasoning via Token-Selective Dual Knowledge Distillation.
ICLR 2026.

Kim, Y., Shin, D., Kang, M., Na, B., & Moon, I. C. (2025).
Distillation of Large Language Models via Concrete Score Matching.
ICLR 2026.

Dao, Q. P., Nguyen, H. S., Pham, K. C., Nguyen, T., Ngo Van, L., Nguyen, D. T.-N., & Le, T. (2026).
SRA: Span Representation Alignment for Large Language Model Distillation.
ACL 2026.

Pham, K. C., Dao, Q. P., Nguyen, T., Ngo Van, L., Le, T., & Nguyen, T. (2026).
MTA: Multi-Granular Trajectory Alignment for Large Language Model Distillation.
ACL 2026.

References IV

Dao, Q. P., Nguyen, H. S., Pham, K. C., Ngo Van, L., Nguyen, D. T.-N., Nguyen, T. H., & Le, T. (2026). **TALAS: Teacher-Anchored Layer Alignment with Adaptive Sharpness-Aware Minimization for Embedding Distillation**.
ACL 2026.