Target-Conditioned Gating for Many-to-Many Argument Assignment in Document-Level Event Extraction

Junru Liao[1]*, Haoran Wang[1]*, Xuanjing Huang[2], Yatian Shen[3]†
[1]International School of Technology, Henan University, Zhengzhou 450046, China
[2]College of Computer Science and Artificial Intelligence,
Fudan University, Shanghai 200441, China
[3]School of Computer and Information Engineering, Henan University, Kaifeng 475004, China
liaojunru@henu.edu.cn, HaoranWang@henu.edu.cn
xjhuang@fudan.edu.cn, ytshen@henu.edu.cn

Abstract

Document-level event extraction often requires the same argument candidate to fill multiple valid event-slot targets, with each assignment grounded in different evidence. Many existing methods introduce target-specific signals during event construction, matching, or role scoring, while explicit per-target argument views jointly conditioned on event-slot semantics and instance-specific information before role scoring remain less explored. This late arrival of target-specificity is especially problematic for many-to-many (M2M) assignment, where the same candidate must preserve distinguishable evidence for several valid targets before decoding. We propose Target-conditioned Gating for Inter-target Decoupling (TGID), an encoding-stage MoE framework that produces target-conditioned argument views before decoding. TGID combines event-slot semantics with instance probes to form target queries and uses independent sigmoid gates to remove direct cross-gate coupling induced by softmax normalization, while schema-structured routing initialization and a multi-positive target-view separation objective stabilize the resulting views. By incorporating instance probes, TGID can distinguish Cross-Type, Cross-Role, and Same-Type Multi-Instance targets before decoding. Across ChFinAnn, DuEE-Fin, and FNDEE, TGID achieves consistent gains over strong trigger-free DEE baselines, especially on M2M subsets. Parameter-matched softmax controls and routing diagnostics indicate that target-sensitive routing contributes beyond additional model capacity. Code is available at https://github.com/LeonaKTH/TGID.

1 Introduction

Document-level event extraction (DEE) aims to convert long, unstructured documents into structured event records by linking arguments scattered across sentences and resolving multiple co-occurring events that often share common entities (Zheng et al., 2019; Yang et al., 2021). In event-dense documents, this creates a recurring structural challenge in which the same argument candidate can simultaneously fill multiple valid event-slot targets, each grounded in a different evidence context, as illustrated in Figure 1.

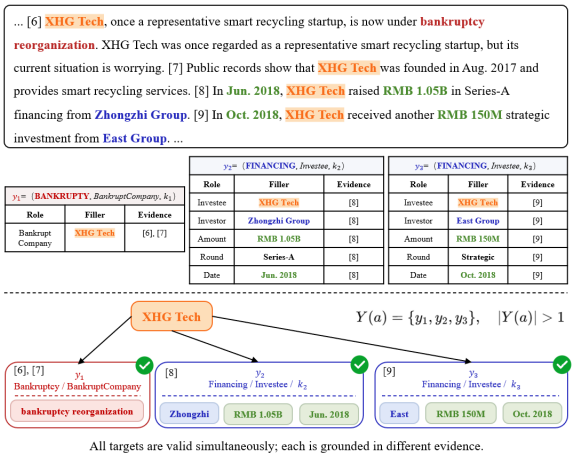


Figure 1: An M2M argument in document-level multi-event extraction. The same argument candidate fills multiple valid assignment targets, each supported by different evidence. Highlighted text marks supporting evidence.

This structure changes the nature of argument assignment. The task is not to select one correct target from competing alternatives, but to preserve evidence for multiple simultaneously valid targets of the same candidate. We call such a candidate an *M2M argument* when $|Y(a)| > 1$, arising as Cross-Type M2M (across event types), Same-Type Multi-Instance M2M (across instances of the same type), or Cross-Role M2M (across roles within overlapping events). These patterns turn DEE from selecting a single best assignment into preserving multiple simultaneously valid event-slot assignments, a setting that is difficult to support with a single

*Equal contribution.
†Corresponding author.

Proceedings of the 15th International Joint Conference on Natural Language Processing and the 5th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 731–746
November 6-10, 2026 ©2026 Association for Computational Linguistics

target-agnostic argument representation.

Prior DEE models have progressively addressed different bottlenecks in multi-event extraction. Graph- and path-based models (Zheng et al., 2019; Xu et al., 2021; Yang et al., 2021; Liang et al., 2022) strengthen cross-sentence argument interaction. Proxy-node methods (Wang et al., 2023; Zhu et al., 2022) introduce event-level representations for global event-set alignment, reducing the reliance on sequential decoding. Probe-based methods (Hu et al., 2025) improve homogeneous event separation through event-specific probes and differential optimization over argument libraries. These approaches address event construction, matching, or instance discrimination at the event or decoder level. Although these methods introduce event- or instance-conditioned representations and interactions, they do not explicitly construct a separate argument view jointly conditioned on event-slot semantics and instance-specific information for each target before role scoring. For M2M arguments, this leaves target-specific distinctions to be modeled primarily during downstream event construction or decoding rather than through explicit per-target argument views. We refer to this mismatch as the late arrival of target-specificity at the level of argument-view construction.

This motivates target-conditioned argument encoding. MoE routing provides a standard mechanism for conditional computation through expert pathways (Shazeer et al., 2017; Fedus et al., 2022; Dai et al., 2024; Huang et al., 2024), but common softmax routing introduces direct cross-gate coupling through competitive normalization. This competition is misaligned with M2M assignment, where different target-conditioned views of the same argument may require different expert combinations.

We propose Target-conditioned Gating for Inter-target Decoupling (TGID), an encoding-stage MoE framework for M2M document-level event extraction. TGID conditions routing on event-slot descriptions and decoder-side instance probes, replaces competitive softmax with independent sigmoid gates, and encodes the same argument as multiple target-conditioned views before event decoding. These views are supplied to downstream event-level decoders, making TGID complementary to existing event-construction and probe-based decoding methods. To make such non-competitive routing stable and discriminative for semantically related targets, TGID further introduces a schema-structured expert specialization prior and a multi-positive target-view separation objective.

This paper makes three contributions:

- We characterize M2M argument assignment as a representation-level encoding challenge across Cross-Type, Cross-Role, and Same-Type Multi-Instance patterns. By combining slot semantics with instance probes, TGID produces target-conditioned routed argument views for Cross-Type, Cross-Role, and Same-Type Multi-Instance targets.
- We propose TGID, an encoding-stage MoE framework that conditions argument routing on event-slot semantics, replaces competitive softmax normalization with independent sigmoid gates, and produces target-conditioned argument views before event decoding, while leaving the downstream decoder unchanged. Schema-structured routing initialization provides the specialization prior needed for stable non-competitive routing, and a multi-positive target-view separation objective further sharpens the resulting target-conditioned views.
- Across ChFinAnn, DuEE-Fin, and FNDEE, TGID shows consistent gains over EPAL, with the largest improvements on M2M subsets. Routing diagnostics via Target Sensitivity (TS), Soft Overlap (SO), and Top-1 Collision (T1C) indicate that target-sensitive expert selection contributes beyond additional model capacity.

2 Related Work

2.1 Document-Level Event Extraction

Document-level information extraction requires models to aggregate evidence across sentences and resolve structured records from long contexts (Zheng et al., 2024); DEE methods address this challenge by constructing event records from document-level candidates through entity-based path expansion, graph interaction, parallel prediction, relation-aware modeling, or pruned complete-graph decoding (Zheng et al., 2019; Xu et al., 2021; Yang et al., 2021; Liang et al., 2022; Zhu et al., 2022). These methods introduce event- or probe-conditioned signals mainly through event construction, matching, or role scoring, while explicit schema-grounded multi-view encoding of each argument before decoding remains less explored.

Recent DEE methods introduce event-level or schema-level conditioning through proxy-node matching (Wang et al., 2023), event-specific probe-library alignment (Hu et al., 2025), schema-description-guided attention (Xu et al., 2024), role-relevance-guided encoding (Liu et al., 2023b), multi-event aggregation (Liu et al., 2024), and iterative parallel generation (Huang et al., 2023). The Same-Type Multi-Instance pattern is also related to event individuation (Gantt et al., 2023) and event co-occurrence-aware argument extraction (He et al., 2023), the challenge of distinguishing co-occurring instances of the same event type within a single document. These methods introduce target-specific signals at different stages, but do not explicitly construct a separate instance-aware argument view for each event-slot target before decoding. Unlike description-guided attention or proxy-node modulation, TGID instantiates a separately expert-routed view of each argument for every instance-aware target before decoding, so that target specificity is carried in the representation itself. Another line of work addresses long-range evidence in document-level event argument extraction through chain reasoning and retrieval augmentation (Liu et al., 2023a; Ren et al., 2023); recent LLM-based methods have explored guided summarization, multi-document extraction, and heuristic-driven prompting in D-EAE and generative event extraction settings (Shuang et al., 2024; Min et al., 2025; Zhou et al., 2024), with more recent work applying reinforcement learning with verifiable rewards to LLM-based EAE (Luo et al., 2026). TGID is orthogonal to these directions: it operates in the trigger-free discriminative DEE setting (Shaar et al., 2025) and addresses the target-conditioning of argument representations rather than evidence enrichment or prompting context.

2.2 Mixture-of-Experts in NLP

Sparse MoE models scale NLP architectures by routing each input to a subset of expert subnetworks (Jacobs et al., 1991; Shazeer et al., 2017; Lepikhin et al., 2020; Fedus et al., 2022). Recent MoE studies improve routing and specialization through expert segmentation, non-competitive expert-choice assignment (Zhou et al., 2022), dynamic routing, dense-to-MoE upcycling, maximum-score routing, and clustered expert specialization (Dai et al., 2024; Huang et al., 2024; Zhu et al., 2024b; Zhang et al., 2025b,a; Dong et al., 2025; Eo et al., 2025). Since expert specialization is not guaranteed by sparse routing alone, recent analyses advocate examining router behavior and expert usage directly rather than relying solely on downstream task accuracy (Lo et al., 2025). Recent work further characterizes expert specialization through collaborative expert interactions (Tang et al., 2026), while routing methods have also been developed to promote more stable expert specialization (Hou et al., 2026).

These advances largely focus on improving expert utilization, routing stability, or load balance for scalable conditional computation. TGID addresses a structurally different problem. For M2M DEE, a key issue is the routing formulation rather than expert capacity alone. Within each target-conditioned gate vector, softmax normalization directly couples gate values through competitive normalization, which may restrict routing sensitivity to different target queries. TGID therefore combines event-slot semantics with instance probes and uses independent sigmoid gates to remove this direct cross-gate coupling, allowing different targets of the same argument to activate different expert subsets.

3 Methodology

3.1 Problem Formulation

Given a document $D = \{w_i\}_{i=1}^{|D|}$, DEE produces structured event records. Let $\mathcal{T}$ be the event type set and $\mathcal{R}_t$ the role set for type $t \in \mathcal{T}$. We distinguish an *event slot* $s = (t, r)$ from an *instance-aware target* $y = (t, r, k)$, where k indexes a specific event instance. The valid target set of argument candidate a is

$$Y(a) = \{y \mid a \text{ is a valid filler for target } y\}, \quad (1)$$

and a is an M2M argument when $|Y(a)| > 1$. In the default setting, TGID constructs an instance-aware target query $q^{(y)} = f([c^{(s)}; p_k])$ for each target $y = (t, r, k)$, where $s = (t, r)$ and p_k is the decoder-side event-specific probe associated with instance k. In the EPAL-style decoder, such probes are constructed from context-enriched entity representations and used to query event-specific argument libraries. This allows Cross-Type, Cross-Role, and Same-Type Multi-Instance targets to obtain distinct routed views before decoding.

Figure 2 shows the overall TGID architecture. Given a target-agnostic argument representation $\tilde{h}_a$, TGID constructs target queries from event-slot descriptions and decoder-side instance probes, acti-

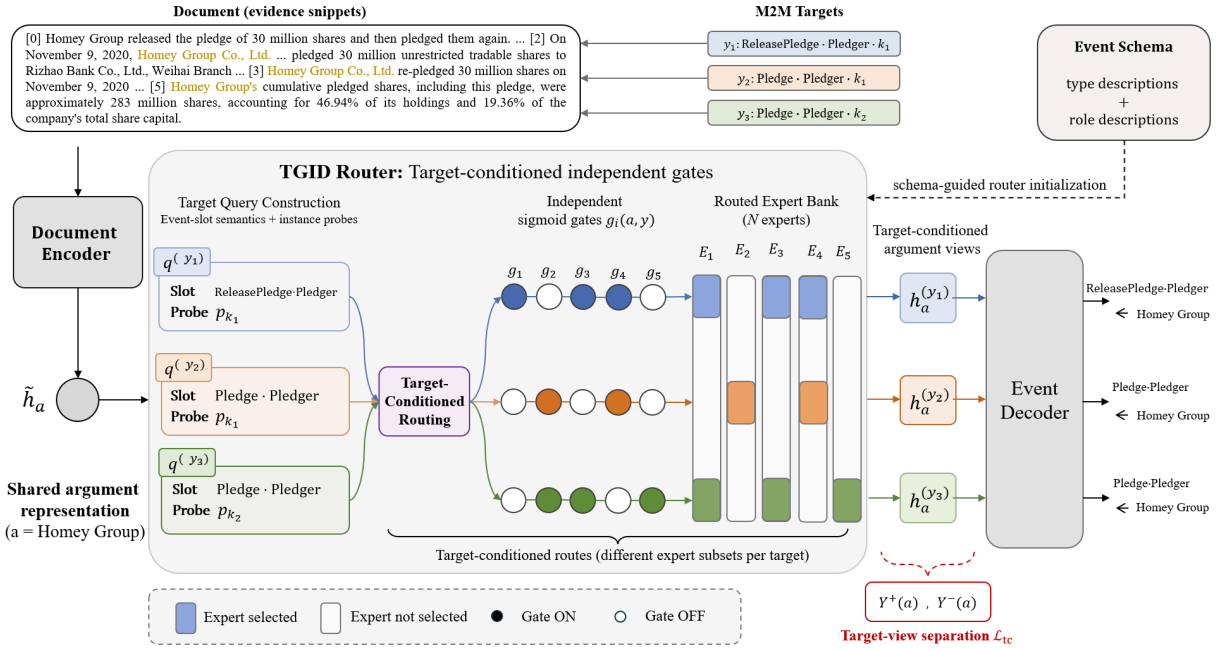


Figure 2: Overall architecture of TGID. Target queries combining event-slot descriptions and instance probes route the same argument through independent sigmoid gates, producing target-conditioned views for downstream event decoding.

vates independent expert gates, and produces target-conditioned views for downstream event decoding.

3.2 Target-Conditioned Independent Routing

Many prior DEE methods introduce event-specific signals through decoder-side querying, matching, or role scoring, but do not explicitly construct a separate argument view jointly conditioned on event-slot semantics and instance-specific information for each target before role scoring. TGID instead routes each argument into target-conditioned expert subspaces using event-slot semantics and decoder-side instance probes.

Target-conditioned routing. For each event slot $s = (t, r)$, we encode its type and role descriptions as a stable target signal:

$$\mathbf{c}^{(s)} = \mathrm{sg}\big(\mathrm{Enc}_d([\mathrm{desc}(t); \mathrm{desc}(r)])_{\texttt{[CLS]}}\big), \quad (2)$$

where $\mathrm{sg}(\cdot)$ is stop-gradient. For target $y = (t, r, k)$, the routing query is generally defined as

$$\mathbf{q}^{(y)} = f\Big([\mathbf{c}^{(s)}; \mathbf{p}_k]\Big), \quad s = (t, r), \quad (3)$$

where $\mathbf{p}_k$ denotes the decoder-side instance probe (Hu et al., 2025) and $f(\cdot)$ is a linear projection. We apply a shared expert E_0 to project $\tilde{\mathbf{h}}_a$ into a routing-compatible base representation, $\mathbf{z}_a = E_0(\tilde{\mathbf{h}}_a)$, and compute per-expert routing scores via bilinear interaction:

$$\phi_i(a, y) = \frac{\mathbf{z}_a^\top \mathbf{W}_i^r \mathbf{q}^{(y)}}{\sqrt{d}}, \quad (4)$$

where $\mathbf{W}_i^r \in \mathbb{R}^{d\times d}$ is expert i's routing matrix. Since the query encodes target semantics, the same candidate can receive different routing scores under different targets.

Independent sigmoid gating. Standard softmax routing induces negatively coupled gates: $\partial g_i/\partial \phi_j = -g_i g_j < 0$ for $i \neq j$. Recent theoretical work on MoE regression further suggests that sigmoid gating can be more sample-efficient than softmax gating under expert-estimation settings, partly due to reduced competition among experts (Nguyen et al., 2024). We do not claim softmax is generally unsuitable for MoE; rather, its competitive normalization is specifically misaligned with M2M event-slot assignment, where multiple target-conditioned views of the same argument must remain simultaneously valid. TGID instead applies independent sigmoid gates:

$$g_i(a, y) = \sigma(\phi_i(a, y)), \quad (5)$$

giving $\partial g_i(a, y)/\partial \phi_j(a, y) = 0$ for $i \neq j$, removing direct cross-gate coupling. The active expert

set is determined by the raw sigmoid gate:

$$\mathcal{E}(a, y) = \{i \mid g_i(a, y) \geq \tau\}, \tag{6}$$

where $\tau = 0.5$ is a fixed threshold; if no gate reaches τ, the top-scoring expert is activated. Active gates are then normalized over the active set. Let

$$Z(a, y) = \mathrm{sg}\left(\sum_{j \in \mathcal{E}(a,y)} g_j(a, y)\right) + \varepsilon.$$

The normalized gate is

$$\tilde{g}_i(a, y) = \frac{g_i(a, y)}{Z(a, y)}, \quad i \in \mathcal{E}(a, y). \tag{7}$$

The stop-gradient denominator stabilizes forward scale while preserving gate-level gradient independence. This independence concerns direct cross-gate derivatives within a fixed active set; the document encoder, E_0, $\{E_i\}$, and $\{W_i^r\}$ remain shared across targets and receive gradients from multiple target losses. The individual experts E_i operate on $\tilde{h}_a$ to produce expert-specific adjustments, which are combined through the target-conditioned gates:

$$\mathbf{h}_a^{(y)} = \mathbf{z}_a + \sum_{i \in \mathcal{E}(a,y)} \tilde{g}_i(a, y) \cdot E_i(\tilde{\mathbf{h}}_a). \tag{8}$$

Different target queries can activate different expert subsets, allowing TGID to produce target-conditioned views for M2M arguments.

Decoder integration and inference-time routing. In the instance-aware setting, TGID instantiates a routed view $h_a^{(y)}$ for each active target $y = (t, r, k)$, where $s = (t, r)$ and p_k is the decoder-side instance probe. During training, active targets are derived from the ground-truth active event types and their decoder-side probes. During inference, active event types are predicted by the downstream decoder and expanded with the corresponding probes. For each candidate argument a and active target y, TGID supplies $h_a^{(y)}$ in place of $\tilde{h}_a$ to the decoder's argument-role scoring module. In our EPAL-style decoder, the probe-label alignment, probe consistency loss, and role-filling procedure remain unchanged, and $\mathcal{L}_{\text{task}}$ is inherited from EPAL without modification. If an active event type is missed during inference, TGID cannot instantiate target-conditioned routed views for that type, as discussed in Limitations. The same representation-level replacement can also be applied to other argument-scoring modules; in Section 4.3, we include a controlled ProCNet-style compatibility check.

3.3 Schema-Structured Expert Specialization

Independent sigmoid gating removes direct cross-gate coupling, but related event-slot targets can still drive experts toward similar representations. We derive schema prototypes $\{\mu_i\}_{i=1}^N$ by grouping event-slot description vectors $\{c^{(s)}\}$ via schema-only clustering (details in Appendix B), and initialize each routing matrix toward a distinct region of the event-slot semantic space. This initialization provides a slot-semantic specialization prior, while the instance probe p_k further refines the target query during routing. Each routing matrix is initialized as:

$$\mathbf{W}_i^r(0) = \mathbf{W}_0 + \alpha\, \boldsymbol{\mu}_i \boldsymbol{\mu}_i^\top, \tag{9}$$

where $\mathbf{W}_0$ is a shared random baseline. The schema prototypes initialize only the routing matrices $\mathbf{W}_i^r$; the expert FFN parameters E_i are not schema-initialized. For semantically similar event types, we apply a small diversity perturbation rather than enforcing orthogonality, preserving shared expert capacity for related types. Together, this provides a schema-structured prior before document-level supervision begins.

3.4 Multi-Positive Target-View Separation

Independent sigmoid gating removes the direct cross-gate coupling induced by softmax normalization, but does not prevent semantically related targets from collapsing toward similar representations, as reflected in the high SO values of sigmoid-only variants (Table 5). Since $q^{(y)}$ encodes the event-slot semantics together with the instance probe each view should reflect, we maximize their alignment via supervised contrastive learning (Khosla et al., 2020). Let $Y^+(a)$ denote the gold target set and $Y^-(a)$ denote schema-compatible non-gold targets (construction details in Appendix B). We define

$$\rho(a, y) = \exp\left(\mathrm{sim}(\mathbf{h}_a^{(y)}, \mathbf{q}^{(y)})/\tau_c\right), \tag{10}$$

$$\mathcal{L}_{tc}(a) = -\log \frac{\sum_{y^+ \in Y^+(a)} \rho(a, y^+)}{\sum_{y \in Y^+(a) \cup Y^-(a)} \rho(a, y)}, \tag{11}$$

where $\mathrm{sim}(\cdot, \cdot)$ is cosine similarity and τ_c is the temperature. All gold targets form the joint positive set; no gold target is used as a negative. $\mathcal{L}_{tc}$ is applied only to M2M arguments with $|Y^+(a)| > 1$, focusing supervision where evidence separation is most needed.

3.5 Training Objective

TGID replaces $\tilde{\mathbf{h}}_a$ with target-conditioned views $\{\mathbf{h}_a^{(y)}\}$ while leaving the downstream task loss $\mathcal{L}_{task}$ unchanged. The overall training objective is:

$$\mathcal{L} = \mathcal{L}_{task} + \lambda \mathcal{L}_{tc}, \tag{12}$$

where λ controls the contrastive weight. A short task-loss warm-up precedes activating $\mathcal{L}_{tc}$; schedule and hyperparameters are in Appendix B.

4 Experiments

4.1 Experimental Setup

Datasets. We evaluate TGID on three publicly available Chinese document-level event extraction benchmarks. (1) **ChFinAnn**[1] (Zheng et al., 2019) is a financial announcement dataset with 32,040 documents and five event types. It contains 71% single-event and 29% multi-event documents, and we follow the standard train/development/test split used in prior work. (2) **DuEE-Fin**[2] (Han et al., 2022) is another financial-domain dataset with about 11,900 documents and 13 event types. It contains 67% single-event and 33% multi-event samples. Since its official test labels are unavailable, we follow previous work (Liang et al., 2022; Wang et al., 2023; Hu et al., 2025) and use the original development set for evaluation while holding out 500 training documents for development. (3) **FNDEE**[3] (Zhu et al., 2024a; Xu et al., 2024) is a military news dataset with 44,200 documents and eight event types. We follow the standard train/development/test split used in prior work. Following the NDEE-style argument extraction setting for trigger-free baselines, we report argument extraction F1 on FNDEE; trigger extraction is not evaluated because TGID follows the trigger-free table-filling formulation. Gold-annotation M2M arguments account for 20.1%, 13.7%, and 18.9% on ChFinAnn, DuEE-Fin, and FNDEE; full statistics are in Table 7.

Evaluation Metrics. Following prior document-level event extraction work (Zheng et al., 2019; Xu et al., 2021; Wang et al., 2023; Hu et al., 2025), each predicted event is matched with the most similar unmatched gold event of the same type. We compute micro-averaged role-level precision (P.), recall (R.), and F1-score (F1); a predicted argument is correct only if both its span and role match the gold annotation. We report overall P./R./F1, as well as F1 on single-event and multi-event documents, denoted as F1(S.) and F1(M.). To evaluate M2M event-slot assignment, we further report F1 on M2M-All, Cross-Type, Same-Type Multi-Instance, and Cross-Role subsets, defined by the target set $Y(a)$ in Section 3.

Baselines. We compare TGID with representative trigger-free DEE methods spanning path-expansion (Doc2EDAG (Zheng et al., 2019); GIT (Xu et al., 2021)), parallel-prediction (DE-PPN (Yang et al., 2021)), relation-modeling (ReDEE (Liang et al., 2022)), proxy-node (ProCNet (Wang et al., 2023)), and probe-library approaches (EPAL (Hu et al., 2025)). Baseline results are taken from prior work when available, with FNDEE scores following Xu et al. (2024) and ProCNet† following the corrected results in Hu et al. (2025). For M2M analysis, EPAL subset scores are recomputed with the same splits, script, and subset definitions as TGID. SEELE (Xu et al., 2024) uses a joint trigger-and-argument paradigm and is not included as a direct baseline. IPGPF (Huang et al., 2023) uses an iterative generation paradigm incompatible with our trigger-free discriminative setting and is likewise excluded. All main TGID variants built on the EPAL-style decoder share the same backbone, entity pipeline, and candidate set; unless explicitly ablated, the remaining components and training settings are held fixed. Softmax-T and Softmax-K are parameter-matched controls: Softmax-T replaces sigmoid gates with target-conditioned softmax over the same experts and routing matrices; Softmax-K additionally applies top-K selection to match TGID's active-expert budget, with schema-structured routing initialization, decoder, backbone, and training schedule held fixed. The **w/o instance probe** variant removes p_k from the target query and uses $\mathbf{q}^{(s)} = f(\mathbf{c}^{(s)})$; **type-only routing** further removes role descriptions from $\mathbf{c}^{(s)}$, isolating the contribution of role-aware slot semantics. To assess compatibility with a structurally different decoder, we additionally integrate TGID with a ProCNet-style proxy-node decoder under the same backbone, entity pipeline, candidate set, and training split. Implementation details are provided in Appendix C.

[1] https://github.com/dolphin-zs/Doc2EDAG

[2] https://aistudio.baidu.com/aistudio/competition/detail/46/0/task-definition

[3] https://www.datafountain.cn/competitions/638/datasets

4.2 Main Results

Overall results on standard benchmarks. Table 1 reports the standard document-level event extraction results on ChFinAnn and DuEE-Fin. TGID achieves the highest overall F1 on both ChFinAnn (86.2) and DuEE-Fin (82.3), outperforming EPAL by 2.8 and 5.9 points, respectively. TGID improves both single-event and multi-event documents. The single-event gains suggest that target-conditioned views also help role disambiguation when several schema roles or candidate fillers are plausible within one document. Multi-event and M2M-specific results provide the cleaner test of the late-arrival bottleneck hypothesis, since these settings place greater demands on target-specific argument disambiguation.

Argument extraction on three benchmarks. Table 2 further reports argument extraction F1 on ChFinAnn, DuEE-Fin, and FNDEE under the NDEE-style trigger-free evaluation setting. TGID improves over EPAL on all three datasets, with gains of 2.8, 5.9, and 2.5 F1 points on ChFinAnn, DuEE-Fin, and FNDEE, respectively; the fine-grained M2M breakdown is analyzed in Section 4.4.

4.3 Ablation Studies

Table 3 reports DuEE-Fin ablations. For the TGID variants in the upper block, the backbone, entity pipeline, and candidate set are fixed; all components and training settings not explicitly ablated are kept unchanged. Softmax-T replaces sigmoid gates with target-conditioned softmax, Softmax-K adds top-K selection, and w/o instance probe removes p_k from $q^{(y)}$ and uses $q^{(s)} = f(c^{(s)})$.

Removing schema-structured routing initialization causes the largest individual drop (-5.8 overall, -11.5 M2M-All), showing that the schema prior is a major component of TGID. The parameter-matched Softmax-T and Softmax-K controls retain the same expert structure and schema initialization but reduce M2M-All F1 to 75.6 and 74.8, respectively, indicating that target-sensitive routing contributes beyond additional model capacity. Type-only routing, removing $\mathbf{p}_k$, and removing $\mathcal{L}_{tc}$ reduce M2M-All F1 by 7.9, 4.0, and 3.7 points, respectively, showing complementary contributions from role-aware slot semantics, instance-aware routing, and target-view separation.

To test decoder compatibility, we integrate TGID with a ProCNet-style proxy-node decoder under the same backbone, entity pipeline, candidate set, and training split. TGID improves this decoder by 3.0 overall F1 and 3.5 M2M-All F1, suggesting that the argument-view replacement is compatible with a structurally different discriminative decoder.

4.4 Analysis

M2M subset analysis. Table 4 reports F1 on M2M subsets across all three benchmarks. The Non-M2M to M2M-All gap is consistent across all three evaluated benchmarks, indicating that M2M difficulty recurs across these settings. TGID improves M2M-All over EPAL by 9.2, 5.4, and 2.3 points on ChFinAnn, DuEE-Fin, and FNDEE, respectively. These M2M-All gains are significant on all three benchmarks under document-level paired bootstrap with 10,000 resamples ($p < 0.05$). Gains on FNDEE are smaller but present across all three subtypes (Cross-Type $+2.6$, Same-Type Multi-Instance $+2.5$, Cross-Role $+3.4$), suggesting that schema-guided specialization may depend on how cleanly event-slot semantics partition the target space. TGID-Softmax-T marginally exceeds TGID on FNDEE Non-M2M (63.8 vs. 63.6), consistent with TGID's advantage concentrating on M2M disambiguation rather than non-M2M arguments. Cross-Type and Cross-Role gains are consistent with routing conditioned on event-slot distinctions, while Same-Type Multi-Instance gains are consistent with instance-aware routing through decoder-side probes. Since Cross-Type arguments are rare in ChFinAnn (N=31; Table 7), its Cross-Type result should be interpreted cautiously; the larger Cross-Type subsets in DuEE-Fin (N=361) and FNDEE (N=811) provide more stable evidence for this setting.

Routing behavior analysis. TGID achieves the highest TS (12.2%) and the lowest T1C (13.7%) in Table 5. The softmax variants show low TS (0.7–1.6%) and high SO (95.0%), indicating nearly identical expert paths for many M2M target pairs. Removing $\mathcal{L}_{tc}$ lowers TS and raises SO and T1C, showing that target-view separation refines routing distinctiveness beyond independent sigmoid gating alone. Removing the instance probe sharply raises T1C to 39.8%, suggesting that same-slot instances tend to collapse to the same dominant expert when routed only by slot semantics. Full TGID reduces SO to 78.7 and T1C to 13.7, providing behavioral evidence of more target-sensitive routing while retaining shared expert mass across co-valid targets.

Model	ChFinAnn					DuEE-Fin				
	P.	R.	F1	F1(S.)	F1(M.)	P.	R.	F1	F1(S.)	F1(M.)
Doc2EDAG	82.7	75.2	78.8	83.9	67.3	67.1	60.1	63.4	69.1	58.7
DE-PPN	83.7	76.4	79.9	85.9	68.4	69.0	33.5	45.1	54.2	21.8
GIT	82.3	78.4	80.3	87.6	72.3	69.8	65.9	67.8	73.7	63.8
ReDEE	83.9	79.9	81.9	88.7	74.1	77.0	72.0	74.4	78.9	70.6
ProCNet†	83.6	78.1	80.8	87.5	73.5	79.3	71.4	75.1	80.1	71.0
EPAL	83.1	83.5	83.4	89.7	76.6	77.3	75.5	76.4	81.2	72.7
TGID	**86.7**	**85.8**	**86.2**	**93.6**	**78.7**	**82.5**	**82.1**	**82.3**	**87.1**	**78.5**

Table 1: Overall precision (P.), recall (R.), and F1-score on ChFinAnn and DuEE-Fin. F1(S.) and F1(M.) denote results on single-event and multi-event documents. Earlier baseline results are taken from prior work when available; † Corrected results from Hu et al. (2025).

Model	ChFinAnn	DuEE-Fin	FNDEE
Doc2EDAG	78.8	63.4	43.9
DE-PPN	79.9	45.1	43.0
GIT	80.3	67.8	48.5
ReDEE	81.9	74.4	56.8
ProCNet†	80.8	75.1	57.3
EPAL	83.4	76.4	58.6
TGID	**86.2**	**82.3**	**61.1**

Table 2: Argument extraction F1 on three document-level event extraction benchmarks under the NDEE-style trigger-free setting. Earlier baseline results are taken from prior work when available; † Corrected results from Hu et al. (2025).

Variant	Overall F1	M2M-All F1
TGID (full)	82.3	79.4
w/ Softmax-T routing	80.0	75.6
w/ Softmax-K routing	77.3	74.8
w/o schema init	76.5	67.9
w/o $\mathcal{L}_{tc}$	79.4	75.7
w/o instance probe	79.9	75.4
type-only routing	77.2	71.5
Additional decoder compatibility check		
ProCNet	75.1	71.8
ProCNet + TGID	78.1	75.3

Table 3: Ablation results on DuEE-Fin. The bottom block tests decoder compatibility with a ProCNet-style proxy-node decoder.

4.5 Practical Analysis

Computational cost. Table 6 compares EPAL and TGID under the same DuEE-Fin evaluation split and profiling protocol. TGID increases the parameter count only modestly (112.79M to 116.11M) and peak inference memory from 3.46 to 3.97 GB, but average inference time rises substantially from 124.13 to 597.37 ms/document. The main practical overhead therefore lies in latency rather than model size or memory.

Expert contribution. Using the same trained TGID checkpoint, we globally mask one routed expert at a time during evaluation without retraining. Masking any expert reduces Overall F1 by 1.07–1.47 points and M2M-All F1 by 0.71–1.59 points, indicating that all four routed experts are functionally used (Table 13). This intervention measures expert contribution, but does not establish a fixed semantic role for individual experts.

Inference robustness. Missed active types affect 11.7% of M2M arguments and 5.9% of M2M target pairs on DuEE-Fin, compared with 26.6% and 18.8% on FNDEE. Replacing predicted active types with oracle types increases the M2M-All gain over EPAL from 5.4 to 5.9 points on DuEE-Fin and from 2.3 to 3.2 points on FNDEE, indicating that active-type recall remains a bottleneck, particularly on FNDEE. Separately, the top-scoring expert fallback in Eq. (6) is triggered for 11.22% of DuEE-Fin target views and affects 10.69% of documents. Documents containing such cases obtain 74.63 F1 versus 83.26 for the remaining documents; we report this as an association rather than evidence that the fallback itself causes the performance gap.

Overall, TGID provides the clearest benefit when M2M assignment is a recurring concern and offline extraction is acceptable, whereas its current inference overhead may limit latency-critical deployment.

Dataset	Model	M2M-All	Non-M2M	C-Type	S-Type	C-Role
ChFinAnn	EPAL	71.6	94.6	67.5	67.1	73.2
	TGID-Softmax-T	76.6	91.2	66.1	74.5	78.1
	TGID-Softmax-K	76.1	90.7	67.4	74.1	77.4
	TGID	**80.8**	**94.9**	**72.3**	**79.0**	**81.6**
DuEE-Fin	EPAL	74.0	82.1	76.0	71.9	66.8
	TGID-Softmax-T	75.6	80.7	76.3	74.7	69.6
	TGID-Softmax-K	74.8	80.1	75.8	73.6	70.3
	TGID	**79.4**	**85.6**	**79.8**	**78.4**	**72.9**
FNDEE	EPAL	58.0	62.7	57.3	57.0	54.6
	TGID-Softmax-T	58.7	**63.8**	58.0	57.6	55.3
	TGID-Softmax-K	59.5	62.5	59.2	58.7	57.4
	TGID	**60.3**	63.6	**59.9**	**59.5**	**58.0**

Table 4: F1 on M2M subsets. C-Type, S-Type, and C-Role denote Cross-Type, Same-Type Multi-Instance, and Cross-Role M2M. EPAL subset scores are recomputed under the same M2M subset definitions. Subset scores are not directly comparable to dataset-level F1 in Table 2.

Model	TS ↑	SO ↓	T1C ↓
TGID-Softmax-T	0.7	95.0	24.6
TGID-Softmax-K	1.6	95.0	26.8
TGID w/o $\mathcal{L}_{tc}$	6.9	83.4	19.1
TGID w/o instance probe	7.8	85.6	39.8
TGID	**12.2**	**78.7**	**13.7**

Table 5: Routing diagnostics on DuEE-Fin M2M arguments (N=4 experts). TS, SO, and T1C denote Target Sensitivity, Soft Overlap, and Top-1 Collision.

Metric	EPAL	TGID
Parameters	112.79M	116.11M
Time (ms/doc)	124.13	597.37
Throughput (docs/s)	8.06	1.67
Peak GPU memory (GB)	3.46	3.97

Table 6: Controlled efficiency profiling on the DuEE-Fin evaluation split (1,020 documents) using the same RTX 3090 and inference protocol.

5 Conclusion

TGID addresses the late arrival of target-specificity in M2M document-level event extraction by generating target-conditioned argument views before decoding. Across three benchmarks, it improves M2M-All and all three M2M subtypes over EPAL, with gains concentrated on M2M structural disambiguation. Parameter-matched softmax controls and routing diagnostics indicate that target-sensitive routing contributes beyond additional model capacity, while ablations indicate complementary contributions from schema-structured routing initialization, instance-aware queries, and target-view supervision. Encoding-stage target conditioning therefore complements event-level construction and probe-based decoding.

Limitations

Two practical limitations arise at inference time. First, TGID depends on active-type recall: if an active event type is missed, target-conditioned views for that type cannot be instantiated, limiting target-specific M2M disambiguation for that event type. Second, target-wise routing introduces additional computational cost. Under controlled DuEE-Fin profiling, average inference time increases from 124.13 ms per document for EPAL to 597.37 ms per document for TGID. The current implementation is therefore better suited to offline extraction than latency-critical deployment. Improving active-type recall and routing efficiency remains future work.

Our evaluation is limited to trigger-free discriminative DEE on Chinese benchmarks, covering financial documents (ChFinAnn and DuEE-Fin) and military-news documents (FNDEE). Although the ProCNet compatibility experiment provides evidence across two discriminative decoder constructions, generalization to trigger-based, generative, English, or multilingual DEE remains untested.

References

Damai Dai, Chengqi Deng, Chenggang Zhao, R.x. Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Y. Wu, Zhenda Xie, Y.k. Li, Panpan Huang, Fuli Luo, Chong Ruan, Zhifang Sui, and Wenfeng Liang. 2024. DeepSeekMoE: Towards ultimate expert specialization in mixture-of-experts language models. In *Proceedings of the 62nd Annual Meeting*

of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1280–1297, Bangkok, Thailand. Association for Computational Linguistics.

Bowen Dong, Yilong Fan, Yutao Sun, Zhenyu Li, Tengyu Pan, Zhou Xun, and Jianyong Wang. 2025. Maximum score routing for mixture-of-experts. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 12619–12632, Vienna, Austria. Association for Computational Linguistics.

Sugyeong Eo, Jung Jun Lee, Chanjun Park, and Heuiseok Lim. 2025. Mixture-of-clustered-experts: Advancing expert specialization and generalization in instruction tuning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 14201–14212, Suzhou, China. Association for Computational Linguistics.

William Fedus, Barret Zoph, and Noam Shazeer. 2022. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39.

William Gantt, Reno Kriz, Yunmo Chen, Siddharth Vashishtha, and Aaron Steven White. 2023. On event individuation for document-level information extraction. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 12938–12958, Singapore. Association for Computational Linguistics.

Cuiyun Han, Jinchuan Zhang, Xinyu Li, Guojin Xu, Weihua Peng, and Zengfeng Zeng. 2022. Duee-fin: A large-scale dataset for document-level event extraction. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 172–183. Springer.

Yuxin He, Jingyue Hu, and Buzhou Tang. 2023. Revisiting event argument extraction: Can EAE models learn better when being aware of event co-occurrences? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12542–12556, Toronto, Canada. Association for Computational Linguistics.

Peixuan Hou, Yunbo Hou, Bin Chen, LI He, Jian Xu, Weiping Li, Bo Zheng, and Guojie Song. 2026. From pseudo-balancing to true specialization: Memory-aware routing for mixture-of-experts. In *Findings of the Association for Computational Linguistics: ACL 2026*, pages 17320–17337, San Diego, California, United States. Association for Computational Linguistics.

Jianpeng Hu, Chao Xue, Chunqing Yu, JiaCheng Xu, and Chengxiang Tan. 2025. Joint learning event-specific probe and argument library with differential optimization for document-level multi-event extraction. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 714–726, Albuquerque, New Mexico. Association for Computational Linguistics.

Guanhua Huang, Runxin Xu, Ying Zeng, Jiaze Chen, Zhouwang Yang, and Weinan E. 2023. An iteratively parallel generation method with the pre-filling strategy for document-level event extraction. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10834–10852, Singapore. Association for Computational Linguistics.

Quzhe Huang, Zhenwei An, Nan Zhuang, Mingxu Tao, Chen Zhang, Yang Jin, Kun Xu, Kun Xu, Liwei Chen, Songfang Huang, and Yansong Feng. 2024. Harder task needs more experts: Dynamic routing in MoE models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12883–12895, Bangkok, Thailand. Association for Computational Linguistics.

Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. 1991. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. *Advances in neural information processing systems*, 33:18661–18673.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. 2020. Gshard: Scaling giant models with conditional computation and automatic sharding. *arXiv preprint arXiv:2006.16668*.

Yuan Liang, Zhuoxuan Jiang, Di Yin, and Bo Ren. 2022. RAAT: Relation-augmented attention transformer for relation modeling in document-level event extraction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4985–4997, Seattle, United States. Association for Computational Linguistics.

Jian Liu, Chen Liang, Jinan Xu, Haoyan Liu, and Zhe Zhao. 2023a. Document-level event argument extraction with a chain reasoning paradigm. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9570–9583, Toronto, Canada. Association for Computational Linguistics.

Wanlong Liu, Shaohuan Cheng, Dingyi Zeng, and Qu Hong. 2023b. Enhancing document-level event argument extraction with contextual clues and role relevance. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 12908–12922, Toronto, Canada. Association for Computational Linguistics.

Wanlong Liu, Li Zhou, DingYi Zeng, Yichen Xiao, Shaohuan Cheng, Chen Zhang, Grandee Lee, Malu

Zhang, and Wenyu Chen. 2024. Beyond single-event extraction: Towards efficient document-level multi-event argument extraction. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 9470–9487, Bangkok, Thailand. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ka Man Lo, Zeyu Huang, Zihan Qiu, Zili Wang, and Jie Fu. 2025. A closer look into mixture-of-experts in large language models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 4427–4447, Albuquerque, New Mexico. Association for Computational Linguistics.

Jianwen Luo, Yongkang Jin, Yu Hong, and Jianmin Yao. 2026. HTMR: Hybrid token masking reinforcement learning with verifiable rewards for event argument extraction with multi-perspective reasoning. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 19853–19873, San Diego, California, United States. Association for Computational Linguistics.

Qingkai Min, Zitian Qu, Qipeng Guo, Xiangkun Hu, Zheng Zhang, and Yue Zhang. 2025. Multi-document event extraction using large and small language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 19254–19285, Suzhou, China. Association for Computational Linguistics.

Huy Nguyen, Nhat Ho, and Alessandro Rinaldo. 2024. Sigmoid gating is more sample efficient than softmax gating in mixture of experts. *Advances in Neural Information Processing Systems*, 37:118357–118388.

Yubing Ren, Yanan Cao, Ping Guo, Fang Fang, Wei Ma, and Zheng Lin. 2023. Retrieve-and-sample: Document-level event argument extraction via hybrid retrieval augmentation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 293–306, Toronto, Canada. Association for Computational Linguistics.

Shaden Shaar, Wayne Chen, Maitreyi Chatterjee, Barry Wang, Wenting Zhao, and Claire Cardie. 2025. Are triggers needed for document-level event extraction? *Transactions of the Association for Computational Linguistics*, 13:1560–1577.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*.

Kai Shuang, Ji Zhou, Qiwei Wang, and Jinyu Guo. 2024. Thinking about how to extract: Energizing LLMs' emergence capabilities for document-level event argument extraction. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 5520–5532, Bangkok, Thailand. Association for Computational Linguistics.

Yuanbo Tang, Naifan Zhang, Yan Tang, Meixuan Chen, Shuhan Huang, Tingyu Cao, and Yang Li. 2026. Specialization through collaboration: Understanding expert interaction in mixture-of-expert large language models. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2326–2339, Rabat, Morocco. Association for Computational Linguistics.

Xinyu Wang, Lin Gui, and Yulan He. 2023. Document-level multi-event extraction with event proxy nodes and hausdorff distance minimization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10118–10133, Toronto, Canada. Association for Computational Linguistics.

Runxin Xu, Tianyu Liu, Lei Li, and Baobao Chang. 2021. Document-level event extraction via heterogeneous graph-based interaction model with a tracker. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3533–3546, Online. Association for Computational Linguistics.

Zijie Xu, Peng Wang, Wenjun Ke, Guozheng Li, Jiajun Liu, Ke Ji, Xiye Chen, and Chenxiao Wu. 2024. Incorporating schema-aware description into document-level event extraction. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, pages 6597–6605. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Hang Yang, Dianbo Sui, Yubo Chen, Kang Liu, Jun Zhao, and Taifeng Wang. 2021. Document-level event extraction via parallel prediction networks. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6298–6308, Online. Association for Computational Linguistics.

Jihai Zhang, Xiaoye Qu, Tong Zhu, and Yu Cheng. 2025a. CLIP-MoE: Towards building mixture of experts for CLIP with diversified multiplet upcycling. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 5406–5419, Suzhou, China. Association for Computational Linguistics.

Xue Zhang, Yunlong Liang, Fandong Meng, Songming Zhang, Yufeng Chen, Jinan Xu, and Jie Zhou. 2025b. Less, but better: Efficient multilingual expansion for LLMs via layer-wise mixture-of-experts. In *Proceedings of the 63rd Annual Meeting of the Association*

for Computational Linguistics (Volume 1: Long Papers), pages 17948–17963, Vienna, Austria. Association for Computational Linguistics.

Hanwen Zheng, Sijia Wang, and Lifu Huang. 2024. A comprehensive survey on document-level information extraction. In *Proceedings of the Workshop on the Future of Event Detection (FuturED)*, pages 58–72, Miami, Florida, USA. Association for Computational Linguistics.

Shun Zheng, Wei Cao, Wei Xu, and Jiang Bian. 2019. Doc2EDAG: An end-to-end document-level framework for Chinese financial event extraction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 337–346, Hong Kong, China. Association for Computational Linguistics.

Hanzhang Zhou, Junlang Qian, Zijian Feng, Lu Hui, Zixiao Zhu, and Kezhi Mao. 2024. LLMs learn task heuristics from demonstrations: A heuristic-driven prompting strategy for document-level event argument extraction. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11972–11990, Bangkok, Thailand. Association for Computational Linguistics.

Yanqi Zhou, Tao Lei, Hanxiao Liu, Nan Du, Yanping Huang, Vincent Y Zhao, Andrew M. Dai, Zhifeng Chen, Quoc V Le, and James Laudon. 2022. Mixture-of-experts with expert choice routing. In *Advances in Neural Information Processing Systems*.

Mengna Zhu, Zijie Xu, Kaisheng Zeng, Kaiming Xiao, Mao Wang, Wenjun Ke, and Hongbin Huang. 2024a. Cmnee: a large-scale document-level event extraction dataset based on open-source chinese military news. *arXiv preprint arXiv:2404.12242*.

Tong Zhu, Xiaoye Qu, Wenliang Chen, Zhefeng Wang, Baoxing Huai, Nicholas Yuan, and Min Zhang. 2022. Efficient document-level event extraction via pseudo-trigger-aware pruned complete graph. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 4552–4558. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Tong Zhu, Xiaoye Qu, Daize Dong, Jiacheng Ruan, Jingqi Tong, Conghui He, and Yu Cheng. 2024b. LLaMA-MoE: Building mixture-of-experts from LLaMA with continual pre-training. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 15913–15923, Miami, Florida, USA. Association for Computational Linguistics.

A Gradient Analysis: Softmax vs. Independent Sigmoid Gating

A.1 Softmax Gating

Under softmax routing, $g_i = \exp(\phi_i)/\sum_j \exp(\phi_j)$, with partial derivatives:

$$\frac{\partial g_i}{\partial \phi_j} = \begin{cases} g_i(1-g_i) > 0, & i = j, \\ -g_i g_j < 0, & i \neq j. \end{cases} \tag{13}$$

For $i \neq j$, the cross-derivative is strictly negative: increasing ϕ_j necessarily decreases the normalized weight of every other expert i.

A.2 Independent Sigmoid Gating

Under independent sigmoid gating, $g_i(a, y) = \sigma(\phi_i(a, y))$. Since $\phi_i(a, y)$ depends only on $\mathbf{W}_i^r$, $\mathbf{z}_a$, and $\mathbf{q}^{(y)}$, and not on ϕ_j for $j \neq i$:

$$\frac{\partial g_i(a, y)}{\partial \phi_j(a, y)} = 0, \quad i \neq j. \tag{14}$$

No normalization links expert gates. This independence holds at the gate computation level; the document encoder, E_0, $\{E_i\}$, and $\{\mathbf{W}_i^r\}$ remain shared across targets, and gradients from different targets still update the same parameters through these shared components.

A.3 Stop-Gradient Normalization Preserves Gate Independence

Under the active-set normalization in Eq. (7), threshold selection in $\mathcal{E}(a, y)$ acts as a non-differentiable routing decision; gradients are not backpropagated through the hard indicator $\mathbf{1}[g_i(a, y) \geq \tau]$. With $Z(a, y)$ treated as a constant during backpropagation, we have:

$$\frac{\partial \tilde{g}_i(a, y)}{\partial \phi_j(a, y)} = \frac{1}{Z(a, y)} \cdot \underbrace{\frac{\partial g_i(a, y)}{\partial \phi_j(a, y)}}_{=0,\, i \neq j} = 0, \quad i \neq j. \tag{15}$$

Gate-level independence is therefore preserved during backpropagation. The stop-gradient denominator still rescales forward activations by the active-set gate sum, so the independence claim applies to routing gradients rather than forward activation magnitudes.

B Implementation Details

B.1 Description and Query Construction

For event slot $s = (t, r)$, the description is formed by concatenating the event type name, a brief type

Dataset	M2M-Doc	M2M-Arg	CT	MI	CR	Avg. $\lvert Y \rvert$
ChFinAnn	44.38	20.12 (N=4454)	0.14 (N=31)	15.05 (N=3332)	6.12 (N=1355)	2.64
DuEE-Fin	31.37	13.70 (N=858)	5.76 (N=361)	9.36 (N=586)	1.61 (N=101)	2.48
FNDEE	44.12	18.86 (N=2030)	7.54 (N=811)	13.81 (N=1486)	3.38 (N=364)	2.38

Table 7: M2M statistics on the test sets. CT, MI, and CR denote Cross-Type, Same-Type Multi-Instance, and Cross-Role; subtype buckets are not mutually exclusive.

Symbol	Meaning	Default
N	number of experts	4
λ	weight of $\mathcal{L}_{tc}$	0.01
τ	raw-gate threshold	0.5
τ_c	contrastive temperature	0.5
α	initialization strength	0.1
ε	normalization constant	10^{-6}
ϵ	perturbation scale	10^{-3}
E_w	warm-up epochs	5

Table 8: TGID hyperparameters used in our implementation. Values are fixed across datasets unless otherwise specified.

description, the role name, and the role description. When role descriptions are unavailable, we use the template: *"The ⟨role⟩ argument in a ⟨type⟩ event."* The concatenated text is encoded by a frozen Chinese RoBERTa-base encoder (Liu et al., 2019); the `[CLS]` representation serves as $\mathbf{c}^{(s)}$. The instance probe p_k denotes the decoder-side event-specific query. In the main experiments, p_k is concatenated with $c^{(s)}$ to construct the instance-aware target query $q^{(y)} = f([c^{(s)}; p_k])$.

B.2 Router Initialization and Hard Negatives

Schema-structured routing initialization. Given $|\mathcal{S}|$ event slots and N experts, we compute pairwise cosine similarity among $\{\mathbf{c}^{(s)}\}$ and assign slots to N groups via a greedy procedure: each group grows by iteratively absorbing the most similar unassigned slot until reaching $\lceil |\mathcal{S}|/N \rceil$ members. The schema prototype $\boldsymbol{\mu}_i$ is the centroid of group G_i:

$$\boldsymbol{\mu}_i = |G_i|^{-1} \sum_{s \in G_i} \mathbf{c}^{(s)}, \tag{16}$$

and the routing matrix is initialized as

$$\mathbf{W}_i^r(0) = \mathbf{W}_0 + \alpha \boldsymbol{\mu}_i \boldsymbol{\mu}_i^\top. \tag{17}$$

This procedure uses schema descriptions only and accesses no document-level annotations. For event-type pairs with cosine similarity ≥ 0.85, we add a perturbation $\boldsymbol{\delta} \sim \mathcal{N}(0, \epsilon^2 \mathbf{I})$ to reduce early routing redundancy without forcing related types into orthogonal subspaces.

Hard negative construction. For M2M argument a with gold targets $Y^+(a)$, hard negatives $Y^-(a)$ are drawn from four sources:

1. Same-type wrong-role: (t, r', k') where t is a gold type but $r' \notin \{r : (t, r, k) \in Y^+(a)\}$;
2. Same-role wrong-type: (t', r, k') where r is a gold role but t' is not a gold type for a;
3. Same-document non-gold: target candidates in the same document that are not in $Y^+(a)$;
4. Semantically similar: target candidates whose event types have cosine similarity ≥ 0.7 to a gold type but are not gold targets.

All targets in $Y^+(a)$ are treated only as positives; no gold target is used as a negative.

B.3 Training Schedule and Hyperparameters

Following prior document-level event extraction work (Wang et al., 2023; Hu et al., 2025), we adopt the Chinese RoBERTa-base encoder (Liu et al., 2019) and optimize with Adam (Kingma and Ba, 2014), using a learning rate of 1×10^{-5} for pre-trained parameters and 1×10^{-4} for randomly initialized parameters, with a batch size of 32. We train for up to 50 epochs on a single NVIDIA RTX 3090 GPU and select checkpoints by validation F1; the mean training time per epoch is approximately 49, 10, and 12 minutes on ChFinAnn, DuEE-Fin, and FNDEE, respectively.

Training proceeds in two phases. During the *warm-up phase* (first E_w epochs), the model is optimized with $\mathcal{L}_{task}$ only, $\mathcal{L}_{tc}$ is inactive, and τ remains fixed at 0.5. Schema-structured routing initialization is applied once before training. During the *joint phase*, $\mathcal{L}_{tc}$ is activated while τ remains fixed.

Note on load balance. TGID does not include an explicit load-balance loss. In DEE, routing operates over schema-defined event slots combined with a bounded set of decoder-side instance probes, rather than an open token stream, and

Model	WB	FL	BA	BB	CF	CL	SD	SI	SR	RT	PR	PL	EC
Doc2EDAG	60.0	78.3	50.6	40.1	63.2	51.5	50.7	52.9	83.7	51.2	64.8	61.7	51.2
DE-PPN	50.7	62.7	41.3	21.4	36.3	23.0	32.9	31.3	67.8	25.8	42.1	36.3	23.4
GIT	58.8	77.6	56.6	44.7	68.5	55.1	58.8	71.2	86.4	45.0	66.4	71.3	53.8
ReDEE	72.2	81.2	58.9	53.4	76.7	56.7	68.2	56.6	90.6	49.9	75.0	77.8	56.6
ProCNet†	75.9	85.1	67.5	61.2	77.0	49.6	69.5	60.5	90.0	60.3	75.5	77.1	**62.0**
EPAL	76.3	86.1	68.3	69.3	76.1	**59.1**	74.5	57.5	91.3	71.1	76.8	**78.2**	61.2
TGID	**78.7**	**87.1**	**70.4**	**71.6**	**80.6**	58.5	**76.3**	**71.4**	**92.0**	**71.3**	**78.3**	76.1	58.1

Table 9: F1 on 13 event types of DuEE-Fin. † Corrected results from Hu et al. (2025).

Model	EF	ER	EU	EO	EP
Doc2EDAG	70.2	87.3	71.8	75.0	77.3
DE-PPN	73.5	87.4	74.4	75.8	78.4
GIT	73.4	90.8	74.3	76.3	77.7
ReDEE	74.1	90.7	75.3	78.1	80.1
ProCNet†	71.4	92.5	67.3	66.7	79.8
EPAL	74.8	93.4	76.3	77.3	81.5
TGID	**78.1**	**94.5**	**80.7**	**78.2**	**83.7**

Table 10: F1 on 5 event types of ChFinAnn. † Corrected results from Hu et al. (2025).

the schema-structured routing initialization assigns semantically distinct slot groups to distinct experts. Routing diagnostics show a T1C of 13.7%, lower than the random-uniform top-1 reference of $1/N{=}25.0\%$ for $N{=}4$ (Table 5), suggesting that valid M2M targets tend to avoid selecting the same dominant expert.

C Additional Experimental Results

ProCNet-style decoder integration. ProCNet already performs event-level conditional aggregation through proxy-specific graph updates. Our compatibility experiment therefore does not treat ProCNet as target-agnostic; it tests whether TGID's argument-view replacement can be inserted before a proxy-node decoder.

Let $\tilde{\mathbf{h}}_{e_k}$ be the entity representation used by ProCNet and $\mathbf{h}_{z_i}$ the representation of proxy node z_i. In the original ProCNet scorer, the role of entity e_k under proxy z_i is predicted from $[\mathbf{h}_{z_i}; \tilde{\mathbf{h}}_{e_k}]$. We construct a proxy-conditioned query $\mathbf{q}^{(i)} = W_q \mathbf{h}_{z_i}$ and use the same TGID expert bank to produce a proxy-conditioned entity view $\mathbf{h}_{e_k}^{(i)}$. This replacement differs from ProCNet's continuous proxy modulation by introducing an explicit expert-subspace view for each proxy-conditioned argument. The updated role scorer replaces only the entity-side input:

$$\mathbf{p}_{i,k} = \mathrm{softmax}\Big(\mathrm{MLP}([\mathbf{h}_{z_i}; \mathbf{h}_{e_k}^{(i)}])\Big). \tag{18}$$

To isolate the representation-replacement effect, we keep ProCNet's native training objective unchanged and do not introduce the TGID-specific target-contrastive loss $\mathcal{L}_{tc}$. The proxy-node decoder, Hausdorff matching objective, null-event padding, and post-processing rules remain unchanged.

Routing metric definitions. For an M2M argument a with valid target set $Y(a)$ and $|Y(a)| > 1$, let $\tilde{\mathbf{g}}(a, y) = [\tilde{g}_1(a, y), \ldots, \tilde{g}_N(a, y)]^\top$ denote the stop-gradient-normalized gate vector. For directional comparison we define the ℓ_2-normalized gate vector $\bar{\mathbf{g}}(a, y) = \tilde{\mathbf{g}}(a, y)/\|\tilde{\mathbf{g}}(a, y)\|_2$. We define three routing diagnostics: **Target Sensitivity (TS)**, **Soft Overlap (SO)**, and **Top-1 Collision (T1C)**. All diagnostics are computed over unordered target pairs (y, y') from $Y(a)$:

$$\mathrm{TS}(a) = \binom{|Y(a)|}{2}^{-1} \sum_{y<y'} \big(1 - \bar{\mathbf{g}}(a, y)^\top \bar{\mathbf{g}}(a, y')\big), \tag{19}$$

$$\mathrm{SO}(a) = \binom{|Y(a)|}{2}^{-1} \sum_{y<y'} \sum_{i} \min\big(\tilde{g}_i(a, y),\, \tilde{g}_i(a, y')\big) \tag{20}$$

$$\mathrm{T1C}(a) = \binom{|Y(a)|}{2}^{-1} \sum_{y<y'} \mathbf{1}\Big[\arg\max_i \tilde{g}_i(a, y) = \arg\max_i \tilde{g}_i(a, y')\Big]. \tag{21}$$

They correspond to Target Sensitivity (TS) (↑), Soft Overlap (SO) (↓), and Top-1 Collision (T1C) (↓), respectively. Dataset-level values are averaged over all M2M arguments in the test set.

Variant	Overall	M2M-All
TGID	82.3	79.4
w/o sg in denom.	80.3	76.6

Table 11: Ablation of stop-gradient normalization in Eq. (7) on DuEE-Fin.

#Experts	2	4	6	8
F1	81.9	82.3	81.7	80.1

Table 12: Expert-number sensitivity on DuEE-Fin.

Stop-gradient normalization ablation. We remove the stop-gradient operation from the denominator in Eq. (7), while keeping the sigmoid gates and normalization form unchanged. The ablation uses the same DuEE-Fin setting as TGID.

The drop indicates that gradient coupling reintroduced through the denominator degrades routing quality, confirming that gate-level independence in backpropagation is distinct from and complementary to the forward scale stabilization that the normalization form itself provides.

Expert-number sensitivity. We vary the number of experts on DuEE-Fin while keeping other settings fixed. As shown in Table 12, F1 peaks at $N = 4$ and does not monotonically increase with more experts, suggesting that TGID's gain is not a simple capacity effect.

Expert masking. To quantify the functional contribution of individual routed experts, we use the same trained TGID checkpoint and globally mask one routed expert at a time during evaluation. No retraining is performed, and all other inference procedures remain unchanged.

Masking any routed expert reduces both Overall and M2M-All F1, showing that all four experts are functionally used in the trained model. This intervention measures expert contribution but does not establish a fixed semantic role for individual experts.

Profiling details. EPAL and TGID are profiled on the same RTX 3090 using the DuEE-Fin evaluation split of 1,020 documents and the same inference protocol. Mean training time per epoch is 37/49 min on ChFinAnn, 7/10 min on DuEE-Fin, and 9/12 min on FNDEE for EPAL/TGID, respectively.

M2M argument statistics. Table 7 reports dataset-level M2M statistics. M2M-Doc and M2M-Arg denote the proportions of documents containing at least one M2M argument and gold arguments satisfying $|Y(a)| > 1$, respectively. CT, MI, and CR give the breakdown across the three M2M subtypes.

Setting	Overall F1	M2M-All F1
Full TGID	82.30	79.40
Mask E_1	80.83	77.81
Mask E_2	81.20	78.69
Mask E_3	80.96	78.41
Mask E_4	81.23	78.49

Table 13: Evaluation-time expert masking on DuEE-Fin. Each routed expert is masked separately without retraining.

Oracle active-type analysis. To quantify the impact of missed event types during inference, we compare the standard predicted-T_{active} setting with an oracle setting that uses gold active event types while keeping the remaining decoding procedure unchanged.

Arg-FB and Tgt-FB denote the proportions of M2M arguments and M2M argument-target pairs associated with missed active event types, respectively. TGID Gain is the normal-setting M2M-All improvement over EPAL in Table 4; Oracle Gain is the additional gain from gold active event types. The smaller Oracle Gain indicates that event-type recall constrains TGID but does not explain away its M2M gains.

Per-event-type results. Tables 9, 10, and 16 report per-event-type F1 to provide a finer-grained view of where TGID improves or degrades. Abbreviations follow prior work (Hu et al., 2025; Xu et al., 2024).

D Case Study

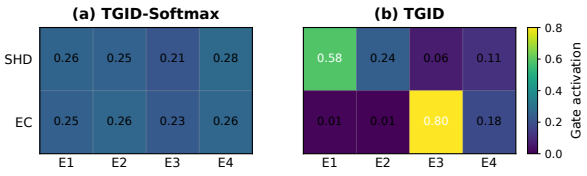


Figure 3: Routing heatmaps for a Cross-Type M2M argument with SHD and EC as valid targets. TGID produces more target-specific expert activations than TGID-Softmax.

Figure 3 visualizes a Cross-Type M2M argument shared by SHD and EC targets. TGID-Softmax assigns nearly identical expert patterns to the two

Event / Role	Gold	EPAL	TGID-Softmax	TGID
Success: Same-Type Multi-Instance M2M in a Pledge document				
Pledge 2 / Pledged Ratio in Holdings	37.50%	– ×	– ×	37.50% ✓
Pledge 3 / Pledged Ratio to Total Shares	7.47%	– ×	– ×	7.47% ✓
Pledge 3 / Pledged Ratio in Holdings	82.87%	– ×	– ×	82.87% ✓
Pledge 4 / Pledged Ratio to Total Shares	17.41%	– ×	– ×	17.41% ✓
Failure: Cross-Type M2M with missed EC type at inference				
EC / Affiliated Company	Zhuoyue Edu.	– ×	Zhuoyue Edu. ✓	– ×[†]
EC / Executive Name	Xu Wenhui	– ×	2018-06-13 ×	– ×[†]
SHD 1 / Stock Abbreviation	Zhuoyue Edu.	2020-08-26 ×	2020-08-26 ×	2020-08-26 ×

[†] EC absent from $\hat{\mathcal{T}}_{\text{active}}$; TGID generates no target-conditioned views for EC targets and reverts to z_a for all EC-role arguments.

Table 14: Case study on DuEE-Fin. The top block shows a Same-Type Multi-Instance success case; the bottom block shows a Cross-Type failure caused by a missed EC event type. ✓ indicates correct extraction and × indicates missing or wrong extraction.

Dataset	Arg-FB	Tgt-FB	TGID	Oracle
DuEE-Fin	11.7	5.9	+5.4	+0.5
FNDEE	26.6	18.8	+2.3	+0.9

Table 15: Effect of missed active event types on M2M arguments. Oracle reports the additional gain from gold active event types.

Model	EP	MA	DE	SU	AC	EH	CO	IN
Doc2EDAG	53.4	26.2	30.5	40.2	53.3	47.9	44.0	50.6
DE-PPN	50.8	25.8	30.6	44.4	42.6	46.2	35.4	32.8
GIT	54.2	27.6	41.0	44.9	51.8	58.2	43.1	56.6
ReDEE	56.6	40.4	50.2	62.4	56.3	63.2	61.0	**74.0**
ProCNet	53.9	44.7	55.8	64.7	54.3	67.7	50.6	62.6
EPAL	54.2	45.1	56.3	65.8	55.1	65.2	**56.4**	62.1
TGID	**58.0**	**48.6**	**56.6**	**69.3**	**57.9**	**66.9**	55.3	62.1

Table 16: F1 on 8 event types of FNDEE.

targets, whereas TGID routes them to distinct dominant experts, consistent with the lower T1C and higher TS in Table 5.

Table 14 gives one success case and one failure case. In the Same-Type Multi-Instance case, EPAL and TGID-Softmax miss all percentage fillers, while TGID recovers all four role-specific values. With instance-aware target queries, same-slot instances can receive different routed views through their decoder-side probes. This allows TGID to separate Same-Type Multi-Instance targets before role scoring, while the downstream probe-label decoder completes event-instance assignment.

The failure case illustrates TGID's dependence on active event-type recall. When EC is absent from $\hat{\mathcal{T}}_{\text{active}}$, no target-conditioned views are instantiated for EC targets, and EC-role arguments fall back to z_a, producing missing outputs. The incorrect date prediction for SHD *Stock Abbreviation* further reflects an upstream entity-recognition error, which is outside the routing module.