PD^3: A Framework for Project Duplication Detection via Adapted Multi-Agent Debate

Anonymous ACL submission

Abstract

Project duplication detection is critical for project quality assessment by preventing investing in newly proposed projects whose topics are already studied. Existing methods rely on word- and sentence-level comparison or solely apply large language models, lacking valuable insights generation and in-depth comprehension of core content and detection criteria. To address this, we propose **PD**^{3}, a framework for **P**roject **D**uplication **D**etection via adapted multi-agent **D**ebate. Inspired by real-world expert discussion and tournament formats, PD^3 employs a fair round-robin competition format in multi-agent debate to retrieve relevant projects and generates both qualitative and quantitative feedback for greater practicality. Over 800 real-world power projects spanning 22 specialized fields are used for evaluation, demonstrating 8.37% and 8.00% improvements in precision and accuracy in two key tasks. Furthermore, we develop and deploy ***Review Dingdang***, an online platform assisting power experts, which has already saved 13.44 million USD across 442 newly proposed projects. Codes are available in this repository.

1 Introduction

Project duplication detection is important for quality assessment, comparing newly proposed projects with reference projects to avoid redundant research investments. Recently, research and development funding and project volumes continuously grow. For example, the State Grid Corporation of China (SGCC) invests over 5.25 billion USD in science projects, obtaining 8,521 authorized patents (The State Grid Corporation of China, 2024). The Smart Grid Grants funding is established with 300 billion USD for projects that improve the power system (The U.S. Department of Energy, 2023). As a result, the expansion increases the need for accurate comparison, making manual inspection difficult and unsustainable. Simultaneously, both reviewers and applicants seek more interpretable feedback, showing dissatisfaction with the simple numerical results provided by general detection methods.

The primary objective of project duplication detection is to retrieve the most relevant reference projects. An ideal method requires a deep understanding of the semantics of the project and domain knowledge, with recall capabilities tailored to domain-specific criteria. Traditional approaches include word frequency-based methods (e.g., ROUGE (Lin, 2004), BM25 (Robertson et al., 2009)) and vector distance-based methods (e.g., Bert (Devlin et al., 2019), gte (Li et al., 2023b)). Word frequency-based methods use word occurrences to represent duplication among texts. Over-reliance on tokenization makes such methods vulnerable to synonym substitution and other text manipulation, and thus ineffective in intentional duplication avoidance cases. Vector distance-based methods encode text semantics as vectors and calculate similarity through functions such as cosine. Some embedders (e.g., gte, bge (Chen et al., 2024a; Li et al., 2023a)) further employ prefix-guided pre-training to enhance task alignment. However, along with the word frequency-based methods, their uniform processing of all texts and highly compressed presentation prevent them from prioritizing core content, lacking more detailed semantic expressions. Due to their inflexibility, these methods are primarily suited for preliminary retrieval.

For further fine-grained retrieval, Large Language Models (LLMs) offer a direct solution. For tasks requiring deep semantic comprehension, like peer review, LLMs can serve as expert reviewers to leverage their understanding capabilities (Shen et al., 2022; Zhang et al., 2025a). Current LLM-based retrieval approaches typically either (1) utilize LLMs as judges to decide the ranking through point-wise scoring or pair-wise comparison (Zheng et al., 2023b), or (2) employ chain-of-thought (CoT) prompting or use LLMs with reasoning abil-

ity (Wei et al., 2022; Guo et al., 2025) to enable task-aware data sorting (Chen et al., 2025b). Benefiting from test-time scalability, these methods overcome rigid word or vector dependencies. Although the aforementioned point-wise scoring and pair-wise comparison methods advance the field, project duplication detection remains challenging.

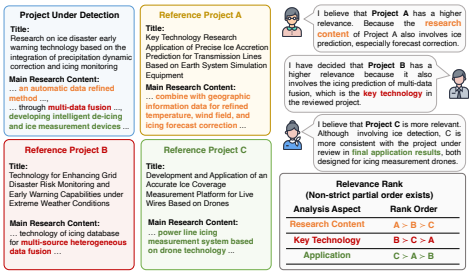

Figure 1: A case where strict partial order is violated. Manual detection process indicates that 3 reference projects are each more relevant from different angles. Multi-angle duplication assessment disrupts strict partial order, rendering point-wise or pair-wise comparisons ineffective for determining duplicate rankings.

From a scenario perspective, **how to retrieve reference projects without strict partial order is challenging**. During duplication detection, domain experts are requested to conduct comprehensive assessments of similarity across multiple orthogonal dimensions (e.g., research content, core methodologies, and application). This process inherently disrupts the strict partial order (where A $\succ$ B and B $\succ$ C $\Rightarrow$ A $\succ$ C) that typically characterizes the similarity between reference projects and the project under detection. As shown in Figure 1, distinct reference projects may demonstrate superior similarity in different aspects, making it difficult even for experienced experts to establish a consistent ranking through conventional point-wise or pair-wise comparison approaches. This indicates that the performance of the existing methods mentioned above is limited by the assumption of the existence of a partial order relationship within the dataset. This observation underscores that more global information on reference projects is required.

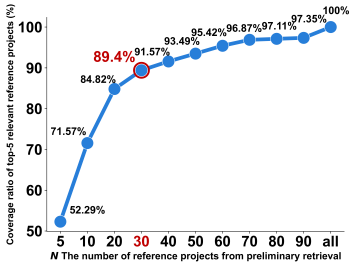

Figure 2: Top-5 coverage with varying retrieval numbers

From a methodological viewpoint, **challenge lies in effectively incorporating the global information of reference projects while overcoming associated long-context difficulties**. Aiming to leverage global information to avoid problems arising from the absence of a strict partial order, single-LLM approaches are however constrained by monolithic analytical perspective. Inspired by the society of mind (Minsky, 1986), multi-agent collaboration like Multi-Agent Debate (MAD) (Liang et al., 2024; Du et al., 2023; Xiong et al., 2023) are proposed. By simulating deliberative discussions among multiple LLM agents, MAD integrates global information and diverse reasoning perspectives. A more fundamental issue, however, arises when processing large candidate sets. As evidenced by Figure 2, covering 90% of human-annotated top-5 relevant projects requires retrieving at least 30 candidates—a global information scale at which LLM performance degrades due to the "lost-in-the-middle" attention phenomenon (Liu et al., 2023). With aggressive reduction risking prematurely discarding relevant projects, narrowing down candidate set by preliminary retrieval fails to resolve the trade-off.

To overcome these challenges, we propose **PD**^{3}, a framework for **P**roject **D**uplication **D**etection via adapted multi-agent **D**ebate. Through an adapted novel round-robin divide-and-conquer MAD mechanism, PD^3 optimizes context length by limiting concurrent projects comparison while maintaining essential global information. This balanced approach enables more accurate retrieval of the top-K most relevant reference projects. Besides, as an insightful framework, PD^3 generates quantitative duplication scores and additionally qualitative output as interpretable supplement for the numerical result. This dual feedback approach helps experts verify results efficiently while providing applicants with clear guidance for project refinement.

Building upon PD^3, we developed an online platform called *Review Dingdang* to help detect duplicate scientific projects in power system. In 2025, the online platform demonstrates significant efficacy, enabling the prevention of approximately 13.44 million USD from being invested in duplicate projects. Our key contributions include:

- We propose **PD**^{3}, a novel and, to our knowledge, the first LLM-based framework for project duplication detection via adapted multi-agent debate.
- In our framework, we introduce **a novel round-robin competition format** in the MAD-based

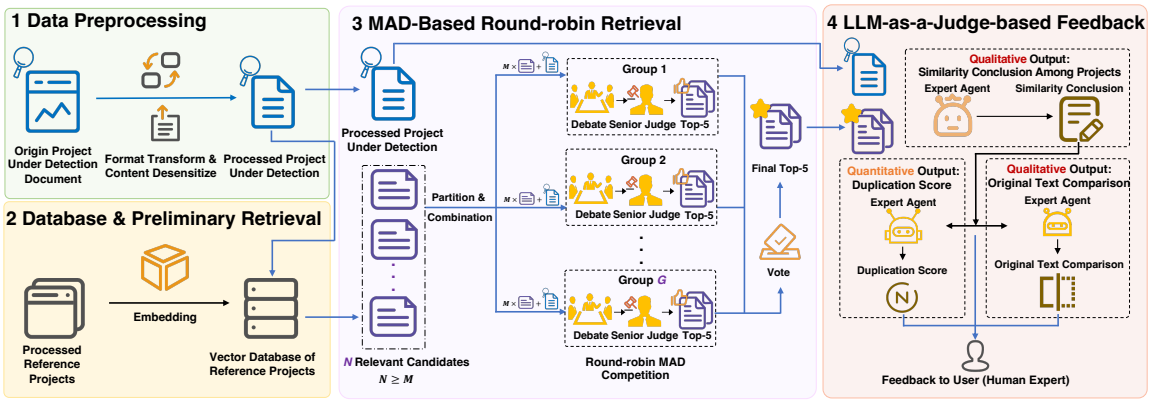

Figure 3: Overview of PD^3 framework. As the I/O sequence of the detection, it has 4 modules: data pre-processing, database and preliminary retrieval, MAD-based round-robin retrieval, and LLM-as-a-Judge-based Feedback.

retrieval method, enabling fairer and more comprehensive candidate evaluation. Additionally, we first propose the **LLM-based dual feedback** design for project duplication detection, integrating both quantitative and qualitative analysis to enhance human collaboration and support.

- Through validation with real-world power project data and cross-domain involving multi-professional domains, PD^3 outperforms baseline methods by 8.34% and 8.00% in two downstream tasks. Based on PD^3, we also implement an online platform, ***Review Dingdang***, achieving practical impact by preventing millions of USD in redundant investments during online detection.

2 PD^3 Framework Design

2.1 Overview

Problem Formulation Before presenting the overall framework, we first formalize the key problem. Let P denote a project, with 2 main project duplication detection tasks defined as follows: (1) Retrieve the top-K most relevant reference projects $R = \{P_j^*\}_{j=1}^K \subseteq S$ for a given project under detection P_u, where $S = \{P_i\}_{i=1}^N$ is the candidate set of reference projects. (2) Provide a quantitative comprehensive duplication score $s_u \in \mathcal{D}$ and (optionally) qualitative evaluation against the retrieval result $R = \{P_i\}_{i=1}^K$, where $\mathcal{D}$ is the score domain and the score function $f : (P_\text{u}, R) \to \mathcal{D}$ evaluates the overall duplication level of P_u.

As illustrated in Figure 3, PD^3 framework comprises four key components:

(1) Data Pre-processing. To handle heterogeneous project data with varying formats and sensitivity levels, this module standardizes input through format unification, regex-based sensitive information desensitization and text content extraction.

(2) Database and Preliminary Retrieval. This module maintains a vector database (offline updatable) of reference projects. PD^3 inputs the content of the project under detection and gets preliminary retrieved candidate reference projects.

(3) MAD-based Round-robin Retrieval. The module organizes the candidate projects into several round-robin sub-competition tasks. Multiple expert agents debate to select the top-5 candidates per sub-competition and a senior judge makes the group final decision by analyzing debate records. The final top-5 projects are determined through aggregated voting across all sub-competitions.

(4) LLM-as-a-Judge-based Feedback. This module analyzes the project under detection by: (1) generating a similarity conclusion, (2) assigning a quantitative duplication score using an LLM judge (based on detection criteria), and (3) enriching output with a text-comparison agent that highlights specific similar expressions in the original texts. The combined feedback provides both measurable and actionable insights for human experts.

2.2 MAD Based Round-robin Retrieval: Debate Makes the Truth Shine Through

Accurately retrieving the most relevant reference projects is the most important part in detection. High-quality retrieval ensures the reliability of downstream evaluation, while errors may lead to final misjudgment. Inspired by practical expert discussion, we propose an adapted multi-agent debate mechanism. It enables more relevant projects to emerge through several debate rounds, where projects with stronger evidence naturally prevail.

First, given the absence of a strict partial order, we depart from the traditional ranking formulation—typically adopted by pair-wise or point-wise approaches—and instead frame the retrieval task

as a many-to-many selection problem. This formulation avoids the need for strict global one-to-one comparisons, thereby effectively circumventing reliance on a strict partial order.

Simultaneously, unlike generic MAD methods designed for QA tasks (e.g., MMLU (Wang et al., 2024b)), our scenario faces a distinct challenge: it must balance global information against context length constraints. This necessitates a tailored decomposition strategy, not a direct MAD application. To achieve an optimal trade-off, we enhance MAD with a round-robin competition mechanism. Originating from real-world tournaments, it prioritizes fairness over elimination intensity, unlike knockout or double-elimination systems. The round-robin format ensures an equitable comparison among all candidates and is better suited for our task.

In specific, we randomly partition N candidate projects primarily retrieved into J sets of K candidate reference projects each. Then, we organize these sets into G unique K-out-of-M sub-competitions(non-repeating combinations, $G = C_J^{M/K}$), where each sub-competition consists of M reference projects from M/K sets (We take $N = 30$, as inspired by Figure 2, $K = 5$, $J = N/K = 6$, $M = 20$, $M/K = 4$ and $G = C_J^{M/K} = 15$).

In each sub-competition group, independent expert agents first select their own top-K candidates and give brief explanations. They then engage in a structured debate by critiquing proposals, answering questions or revising selections with all prior debate records. After a fixed number of debate rounds, a senior expert agent makes the final group decision based on the discussion record. Once all sub-competitions finish, we aggregate results via majority voting (as studied in (Kaesberg et al., 2025) that voting is more suitable for knowledge-based tasks) to select the global top-K reference projects. Another advantage of this format is that group competitions are independent and can run in parallel, significantly reducing execution time.

In summary, our adapted round-robin MAD-based retrieval is able to deeply analyze project and detection criteria, enabling more accurate horizontal comparison and candidate selection. Compared to other LLM-based methods, this approach offers three key advantages: (1) Eliminating cost uncertainty caused by uncontrolled reasoning length; (2) Better balancing non-strict partial order relationships with context length; (3) Enhancing efficiency through parallel task decomposition, while retaining the benefits of test-time scaling.

2.3 LLM-as-a-Judge Based Feedback: Quantitative and Qualitative Output

Through literature review and expert consultation, we identify another key limitation: lack of constructive feedback. Effective feedback should bridge human-system interaction by delivering actionable insights to improve detection efficiency and project quality. Nevertheless, most methods fail to generate interpretable final duplication scores, while inconsistent scoring ranges or distributions undermine result credibility. Few methods detect duplication beyond continuous word matches, neglecting overlaps in core points defined by detection criteria. This feedback deficiency hampers review efficiency, forcing experts and applicants to rely on rigid duplication thresholds. Consequently, non-core duplication in high-quality projects risks misjudgment, while meritorious projects miss potential improvement opportunities.

To address these limitations, we first develop an LLM-based agent feedback module that delivers comprehensive quantitative and qualitative feedback. For quantitative feedback, we employ a full-project-level LLM-as-a-Judge approach with well-defined criteria (1-10 scale, higher scores indicating greater duplication) and task-criteria-specific prompts. Unlike point-wise or pair-wise scoring, our agent assesses the target project using all top-K relevant reference projects as context, avoiding reliance on strict partial ordinal comparisons while leveraging richer global information. This method also offers flexibility: scoring rules can be easily adapted by modifying the prompt, for instance, instructing the agent to assign a high score if any one reference project is highly relevant.

Besides, we design two key qualitative feedback to enhance interpretability: **(1) Similarity Conclusion.** A human-readable conclusion comparing the project under detection with the top-K relevant reference projects, highlighting similarities in main content, key technologies and application achievements. This overall summary also aids in enhancing quantitative assessment performance when integrated into the input. **(2) Original Text Comparison.** This output identifies semantically similar segments in original text based on the similarity conclusion. Unlike traditional word-based duplication detection, this method effectively detects relevant content while mitigating the effects of deception tactics like word substitution.

To enable better direct human-interaction, this module is designed to prioritize both efficiency and tangible user benefit. By optimizing quantitative outcomes and bridging gaps in qualitative analysis, PD^3 is substantially enhanced to support end users, establishing a more human-centered framework.

3 Experiments

In this section, we present a comprehensive evaluation of PD^3. Our investigation is guided by the following research questions: **RQ1**: How powerful is PD^3 retrieval? **RQ2**: How helpful is PD^3 feedback? **RQ3**: To what extent are all components comprising PD^3 valid to its performance? **RQ4**: How does PD^3 perform in cross-domain fields? **RQ5**: How robust is PD^3 to its hyperparameters?

We address RQ1 and RQ2 through primary experimental analysis in Section 3.3, and discuss RQ3 and RQ4 via ablation study and cross-domain analysis in Sections 3.4 and 3.5. For RQ5, We conduct detailed analysis on base models, the number of candidates in each sub-competition, and the number of agents or debate rounds in Section A.3.

3.1 Dataset

To rigorously evaluate PD^3's real-world performance, we take the power field as an example and analyze a dataset of 833 scientific and technological projects across 3 years (from 2022 to 2024), with an average raw data length of 27.1k words and 14.6k tokens, sourced from SGCC.

These projects focus on new technologies in power systems and cross-application of cutting-edge technologies in other fields, like artificial intelligence. Common topics include AI-based power consumption forecasting, line icing prediction, and carbon emission detection. Encompassing 22 distinct specific domains (e.g., dispatching, digitalization, and informatization), the dataset captures the breadth of modern power systems. Within one project, it often involves both actual scenarios in the power field and technical solutions from other fields, further increasing the complexity of the duplication detection. See details in Section A.2.

3.2 Experiment Settings

Tasks Utilizing the symbols described in Section 2.1, we define two evaluation tasks with expert-annotated project data in the power field.

Task 1: Most relevant top-5 retrieval Given a project under detection P_{u} and the set of 30 candidate reference projects from preliminary retrieval using vector distance-based retrieval from the database $S = \{P_i\}_{i=1}^{30}$. The task outputs a result set $R = \{P_j^*\}_{j=1}^{5} \subseteq S$ consisting of the top-5 items most relevant reference projects.

For cost efficiency, we randomly select 331 projects as test items. Human experts annotate the optimal result $\hat{R} = \{\hat{P}_j^*\}_{j=1}^{5}$ for each test item. We employ a cross-validation style detection setting: when one project is under detection, all other 832 projects serve as reference candidates. This setting is closer to the cross-checking required for the same batch of projects in the real world.

Task 2: Comprehensive duplication score assessment of the project Given two projects under detection $P_{\text{u-A}}$ and $P_{\text{u-B}}$, along with their top-5 relevant reference project sets $R_A = \{P_{Aj}\}_{j=1}^{5}$ and $R_B = \{P_{Bj}\}_{j=1}^{5}$, the task outputs duplication scores $s_{\text{u-A}}, s_{\text{u-B}} \in \mathcal{D}$. Where $\mathcal{D}$ is the score range.

Since different algorithms use distinct score ranges $\mathcal{D}$ and scoring functions with different distributions $f : (P_{\text{u}}, R) \to \mathcal{D}$, we set up task 2 in such a way to ensure fair comparison. For test set, we randomly sample 100 project pairs $C = \{(P_{\text{u-A}j}, P_{\text{u-B}j})\}_{j=1}^{100}$ from 331 human-annotated projects. Three experts independently vote on which project has higher duplication, yielding annotations $\hat{H} = \{H_j\}_{j=1}^{100}$, where $H_j \in \{\text{u-A}, \text{u-B}\}$.

Baselines. For a comprehensive comparison, we evaluate methods from four categories:

Word frequency-based (WF) . ROUGE-L: Measures text similarity via longest common subsequence. BM25: Enhanced TF-IDF approach using term frequency and inverse document frequency.

Vector distance-based (VD) . gte-1.5B (gte-Qwen2-1.5B-instruction): Transformer-based embedder (w/ or w/o instruction). RATSim (Zhang et al., 2024): Embedder for near-duplicate text retrieval. Reranker: Larger embedders (e.g., gte-7B (gte-Qwen2-7B-instruction) (Li et al., 2023b), Qwen3-8B (Qwen3-Embedding-8B) (Zhang et al., 2025b)) or pretrained rerankers (jina (jina-reranker-v2-base-multilingual) (Sturua et al., 2024), bge (bge-reranker-v2-m3)).

LLM-based (LLM) . DeepSeek V3 (Liu et al., 2024a): Open-source LLM that generates responses directly from input. DeepSeek R1 (Guo et al., 2025): Reasoning-enhanced LLM that performs self-critique. LLM-as-a-Judge (Chan et al., 2023; Liang et al., 2024; Du et al., 2023) : Use

Table 1: Experiment results of task 1: Most relevant top-5 retrieval.

Method	Prec@5	Match@K				
		K = 1 (Hit Rate@5)	K = 2	K = 3	K = 4	K = 5
Random **(Random)**	16.80	209 | 63.14	64 | 19.34	5 | 1.51	0 | 0.00	0 | 0.00
ROUGE-L **(WF)**	23.99	248 | 74.92	125 | 37.76	23 | 6.95	1 | 0.30	0 | 0.00
BM25 **(WF)**	28.70	273 | 82.48	145 | 43.81	52 | 15.71	5 | 1.51	0 | 0.00
gte-1.5B **(VD)**	38.13	296 | 89.43	219 | 61.66	97 | 29.31	18 | 5.44	1 | 0.30
gte-1.5B with instruction **(VD)**	37.82	294 | 88.82	213 | 64.35	96 | 29.00	21 | 6.36	2 | 0.60
gte-7B as reranker **(VD)**	38.97	298 | 90.03	212 | 64.05	105 | 31.72	29 | 8.76	1 | 0.30
Qwen3-8B as reranker **(VD)**	40.18	301 | 90.94	220 | 66.47	110 | 33.23	32 | 9.67	2 | 0.60
jina as reranker **(VD)**	27.98	275 | 83.08	141 | 42.60	42 | 12.69	4 | 1.21	1 | 0.30
DeepSeek V3 **(LLM)**	36.86	297 | 89.73	194 | 58.61	88 | 26.59	27 | 8.16	4 | 1.21
DeepSeek R1 **(LLM)**	39.52	298 | 90.03	214 | 64.65	110 | 33.23	29 | 8.76	3 | 0.91
LLM-as-a-Judge **(LLM)**	38.49	303 | 91.54	211 | 63.75	99 | 29.91	21 | 6.34	3 | 0.91
TourRank **(LLM)**	37.04	290 | 87.61	193 | 58.31	100 | 30.21	27 | 8.16	3 | 0.91
MAD Vanilla **(MAD)**	39.64	307 | 92.75	229 | 69.18	129 | 38.97	**42 | 12.69**	3 | 0.91
DMAD **(MAD)**	38.85	295 | 89.12	215 | 64.95	102 | 30.82	31 | 9.37	0 | 0.00
PD^3 MAD Round-robin (Ours)	**44.23**	**310 | 93.66**	**238 | 71.90**	**133 | 40.18**	41 | 12.39	**10 | 3.02**

a single LLM as a judge and evaluate the task through generative or point-wise scoring methods. TourRank (Chen et al., 2025b): LLM-based information retrieval methods with tournament format.

MAD-based (MAD) . MAD Vanilla: Directly apply vanilla MAD for 5-out-of-30 retrieval. DMAD (Liu et al., 2025): Recently proposed MAD combining diverse reasoning approaches and achieving state-of-the-art performance.

Settings In task 1, we employ gte-Qwen2-1.5B-instruction (without instruction) as the embedder for preliminary retrieval to obtain 30 candidate projects. In task 2, for baseline methods that do not directly output direct duplication score, we report both the maximum and average score among top-5 candidates for fair comparison. Additionally, we have two settings using different top-5 reference sets R: **(1) Human Retrieval**: Uniform expert-annotated reference set for direct method comparison. **(2) Method Retrieval:** Method-specific top-5 reference set, assessing end-to-end detection performance. It can be viewed as the overall performance evaluation. Section A.5 provides more details.

Evaluation Metrics In Task 1, we adopt 2 metrics: **Precision@5 (Prec@5)** and **Match@K**. Precision@5 $= |R \wedge \hat{R}|/5$ measures selection overlap with experts. Match@K $= \sum \mathbb{I}(\left|R \wedge \hat{R}\right| \geq K), K = 1, 2, \ldots, 5$ calculates the results that overlap with the expert selection greater than or equal to K. Where $\mathbb{I}(\cdot)$ is the indicator function (1 if true, else 0). Particularly, Match@1 $\equiv$ Hit Rate@5. For Match@K, we additionally report its ratio to the size of test set in the form Match@K | (Match@K/Size of test set).

In Task 2, we use **Accuracy (Acc)** on 100 test sets, with 2 evaluation approaches: **(1)Origin Group (Origin)**: Strict expert majority as ground truth. **(2) Weighted Group (Weighted)**: Expert votes as weights (e.g., 2A:1B $\rightarrow$ B scores 0.33). This is due to that expert disagreement reflects comparison difficulty and multi-dimensional analysis. For clarity, all metrics are reported in percentages.

3.3 Experiment Results

Analysis of Task 1 (RQ1) Table 1 presents the experimental results for Task 1[1]. The round-robin MAD method in PD^3 achieves superior results. Specifically, compared with the overall suboptimal method, MAD Vanilla, it improves Precision@5 by 4.59%, showing the additional gains from adaptive design when both benefiting from the MAD inference capability. Traditional WF and VD methods show limited effectiveness, suggesting these simplistic methods are inadequate for duplicate detection retrieval. Among LLM-based methods, DeepSeek R1 and LLM-as-a-Judge outperform others, indicating that enhanced reasoning during inference can better utilize LLM capabilities. Adopting the tournament design as the core improvement, TourRank failed to demonstrate better performance, revealing that relying solely on multi-round selection with a single LLM is not enough.

The superior performance of MAD-based methods demonstrates their effectiveness in multi-perspective analysis through debate mechanisms. PD^3 performs better than DMAD, which is the

[1]Due to page limitations, we only show some of VD baselines with better performance and other results are in Table 5

MAD method with the state-of-the-art performance. This indicates that the MAD methods for solving general problems cannot be used directly and simply to solve complex practical problems.

Different K settings in Match@K metric further reveal our method's growing advantage with increasing task difficulty (higher K values). Compared to baselines, it achieves average relative improvements of 6.76% (K=1), 22.96% (K=2), 49.95% (K=3), 85.71% (K=4), and 464.89% (K=5). This highlights our method is more competitive and has greater application value in complex scenarios.

Table 2: Experimental results of task 2.

Method	Human Retrieval (Acc) Origin	Human Retrieval (Acc) Weighted	Method Retrieval (Acc) Origin	Method Retrieval (Acc) Weighted
ROUGE-L MAX **(WF)**	62.00	58.67	56.00	57.33
ROUGE-L AVG **(WF)**	64.00	59.33	65.00	62.33
BM25 MAX **(WF)**	53.00	55.00	53.00	55.00
BM25 AVG **(WF)**	53.00	55.00	54.00	55.33
gte-1.5B MAX **(VD)**	58.00	49.00	65.00	51.00
gte-1.5B AVG **(VD)**	**65.00**	54.00	61.00	51.00
reranker MAX **(VD)**	-	-	63.00	49.00
reranker AVG **(VD)**	-	-	62.00	52.00
LLM-as-a-Judge MAX **(LLM)**	50.00	48.67	47.00	45.00
LLM-as-a-Judge AVG **(LLM)**	58.00	57.67	64.00	62.00
PD^3 feedback (Ours)	64.00	**62.67**	66.00	63.00
PD^3 feedback with conclusion (Ours)	-	-	**67.00**	**64.67**

Analysis of Task 2 (RQ2) Table 2 presents the experimental results for Task 2[2]. Our method outperforms all other baselines under all evaluation and retrieval settings, which demonstrates the superiority of PD^3's quantitative feedback.

Under *Human Retrieval* setting, PD^3 achieves average improvements of 6.13% and 8.00% across groups, demonstrating more reasonable quantitative feedback given identical input. The larger gain in the *Weighted Group* particularly indicates that our method is more aligned with human review experts' preferences. Under *Method Retrieval* setting, performance gains increase to 7.00% and 9.00%, confirming that better retrieval enhances final scoring quality and validating PD^3's effectiveness. Notably, with quantitative output as prior knowledge, the advantages expand further to 8.00% and 10.67%, highlighting that intermediate agent processing can optimize quantitative feedback.

Furthermore, we conduct a comparative evaluation with human experts to assess the qualitative feedback of PD^3 against other prevalent duplication detection methods, like CNKI (Zhang and Sun, 2012). From an output perspective, CNKI is limited to providing statement-level duplicate comparisons alongside citations of original text, as well as passage- and full-text duplication scores (based on word frequency). Nevertheless, it lacks in-depth duplicate analysis capabilities and exhibits poor interpretability. For PD^3, expert evaluations consistently indicate that its qualitative feedback not only delivers more comprehensive content but also significantly enhances reviewers' understanding of specific duplication instances. This capability effectively reduces the review workload and assists project applicants in improving quality.

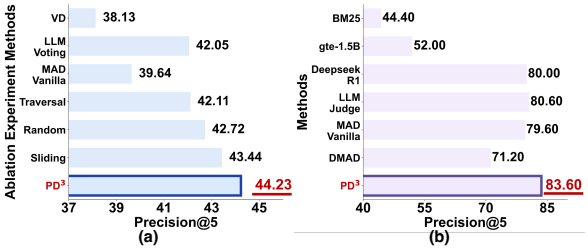

Figure 4: Ablation study experiment results (a) and cross-domain experiment results (b).

3.4 Ablation Study

Ablation Study Settings To evaluate the effectiveness of each component in PD^3 retrieval, we conduct ablation experiments on 4 kinds of variants: (1) *PD^3 w/o MAD* **(VD)**: Removes the entire module, directly using primary retrieval results; (2) *PD^3 w/o MAD, with voting* **(LLM Voting)**: Replaces MAD with one LLM to process sub-competitions (same partition) and get final result through voting; (3) *PD^3 with MAD, w/o round-robin* **(MAD Vanilla)** Remove the round-robin competition format; (4) *PD^3 with MAD in other competition formats* (i) **(Traversal)** Each round selects 5 out of 10, then adds 5 new candidates until completion. (ii) **(Random)** Replace non-repeating combination with a random partition for sub-competitions construction. (iii) **(Sliding)** Replace round-robin with sliding window (step=1).

Ablation Study Analysis (RQ3) Figure 4 (a) presents the ablation results. The full PD^3 demonstrates superior performance over all variants, validating the effectiveness of core components. Besides, the significant performance gap between PD^3 and **VD** confirms the importance of LLM integration in our framework. Simultaneously, the performance of **LLM Voting** decreases, showing that the adapted MAD mechanism contributes more than the voting mechanism alone. Also, all MAD variants with modified competition formats exhibit reduced effectiveness, indicating that the round-robin

[2]In Human Retrieval setting, rerankers' scores are omitted due to using manually annotated top-5 candidates, while PD^3 with conclusion scores are excluded as they represent composite performance only specific to Method Retrieval setting.

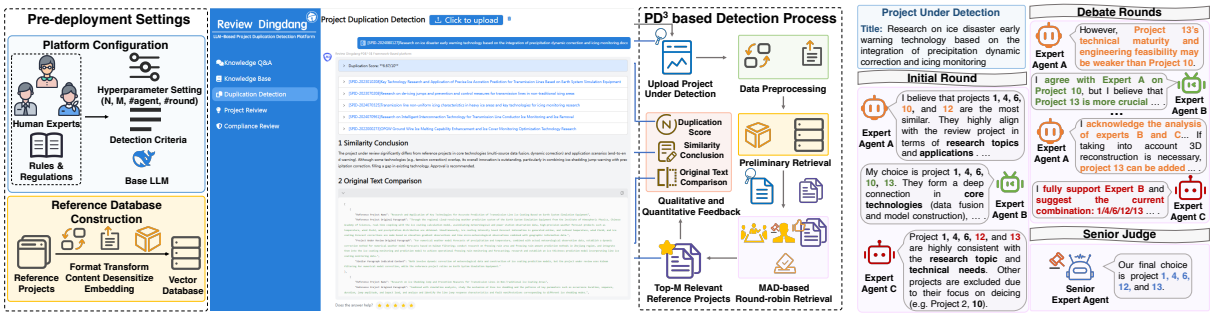

Figure 5: (a) ***Review Dingdang***, the power project duplication platform based on PD^3. Through simple configuration, *Review Dingdang* has been deployed in the review center. It conducts duplicate review of newly proposed projects based on the PD^3 framework and provides sufficient and helpful feedback to human experts. (b) **Case study** on one sub-competition in *Review Dingdang*. 3 experts first independently make their top-5 choices. During debating, Expert A ultimately accepts the views of B and C. A senior judge then makes the final decision.

structure guarantees equal participation opportunities for all candidates, optimally balancing global information capture with context length constraints. Complete results can be found in Table 7.

3.5 Cross-domain Performance

To better explore the duplicate detection performance of PD^3 in other professional fields, we conduct cross-domain experimental research based on bigPatent (Sharma et al., 2019), a diverse and regularly updated dataset of U.S. patent data. We selected 1,000 mechanical patents as a candidate pool and used 5 different outstanding LLMs to rewrite 100 of them to simulate the duplicate detection scenario of non-innovative patents in the real world. The experimental results are shown in Figure 4 (b), where PD^3 still demonstrates the best performance, with a 3.00% improvement over the suboptimal method. This proves the applicability of PD^3 for project duplicate detection across different fields. Detailed dataset processing and experimental results can be found in A.2 and A.6.

4 Application

Platform and Case Study Based on the PD^3, we develop *Review Dingdang*, an online platform for power project duplication detection. As illustrated in Figure 5 (a), the platform architecture supports pre-deployment configuration by domain experts through an intuitive interface. Upon submission of a newly proposed project, the platform automatically performs duplication analysis leveraging PD^3, subsequently presenting comprehensive detection results, including project under detection, relevant reference projects, and quantitative-qualitative feedback. Notably, *Review Dingdang* incorporates a human-in-the-loop mechanism that allows experts to iteratively refine system performance. This is achieved through two primary intervention modalities: (i) dynamic adjustment of debate prompt rules in the PD^3 framework, and (ii) direct denotation feedback on bad cases to optimize the platform's detection algorithms. Figure 5 (b) provides a case study in one sub-competition and a more detailed analysis is provided in Section A.4.

Application Impact With *Review Dingdang*, human experts have conducted duplication detection on **442** newly proposed projects in 2025, attempting to apply the SGCC's scientific fund. During detection, the platform helps experts detect **41** ineligible projects highly duplicated with history projects (9.28%), and save 13.44 million USD, demonstrating its effectiveness and potential for positive social impact. Compared to human-review method used in SGCC before, which requires about 1 hour per project, PD^3 only needs **3 minutes** and **1.34 USD**. Particularly, in live detection test in 2025, *Review Dingdang* detected 24 more highly-duplicated projects (8.09 million USD) and saved over 400 person-hours, a 9-fold reduction on time and hundred times on cost. More details of runtime performance analysis are provided in Section A.4.

5 Conclusion

In this work, we present PD^3, a framework for project duplication detection via adapted multi-agent debate. Its novel round-robin competition MAD-based retrieval method achieves a balance between global information and context length. For the consideration of human-center, PD^3 provides both quantitative and qualitative feedback. Experiments on real-world project data demonstrate its superiority. Besides, our deployed platform *Review Dingdang* already delivers social impact.

Limitations

This work has some limitations. First, the test set size is constrained by expert annotation costs and lack of open-source project data. Second, due to the scarcity of public datasets and the requirement for desensitization of power datasets, we limit the use of text modality for review in our work. Future directions include enhancing detection performance through LLM-based reinforcement learning and expanding to multi-modal analysis.

References

Anthropic. 2025. Introducing claude sonnet 4.5. Accessed: 2025-12-20.

Imene Bensalem, Paolo Rosso, and Salim Chikhi. 2014. Intrinsic plagiarism detection using n-gram classes. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1459–1464.

ByteDance. 2025. volcengine. Accessed: 2025-12-20.

Chi-Min Chan, Weize Chen, Yusheng Su, Jianxuan Yu, Wei Xue, Shanghang Zhang, Jie Fu, and Zhiyuan Liu. 2023. Chateval: Towards better llm-based evaluators through multi-agent debate. *arXiv preprint arXiv:2308.07201*.

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024a. Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. *arXiv preprint arXiv:2402.03216*.

Justin Chen, Swarnadeep Saha, and Mohit Bansal. 2024b. Reconcile: Round-table conference improves reasoning via consensus among diverse llms. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7066–7085.

Xi Chen, Mao Mao, Shuo Li, and Haotian Shangguan. 2025a. Debate-feedback: A multi-agent framework for efficient legal judgment prediction. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 462–470.

Yiqun Chen, Qi Liu, Yi Zhang, Weiwei Sun, Xinyu Ma, Wei Yang, Daiting Shi, Jiaxin Mao, and Dawei Yin. 2025b. Tourrank: Utilizing large language models for documents ranking with a tournament-inspired strategy. In *Proceedings of the ACM on Web Conference 2025*, pages 1638–1652.

Google Deepmind. 2025. Introducing gpt-5. Accessed: 2025-12-20.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B Tenenbaum, and Igor Mordatch. 2023. Improving factuality and reasoning in language models through multiagent debate. In *Forty-first International Conference on Machine Learning*.

Andrew Estornell and Yang Liu. 2024. Multi-llm debate: Framework, principals, and interventions. *Advances in Neural Information Processing Systems*, 37:28938–28964.

Wei Fan, JinYi Yoon, and Bo Ji. 2025. iMAD: Intelligent multi-agent debate for efficient and accurate llm inference. *arXiv preprint arXiv:2511.11306*.

Juraj Gottweis, Wei-Hung Weng, Alexander Daryin, Tao Tu, Anil Palepu, Petar Sirkovic, Artiom Myaskovsky, Felix Weissenberger, Keran Rong, Ryutaro Tanno, and others. 2025. Towards an ai co-scientist. *arXiv preprint arXiv:2502.18864*.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, and others. 2024. A survey on llm-as-a-judge. *arXiv preprint arXiv:2411.15594*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Lars Benedikt Kaesberg, Jonas Becker, Jan Philip Wahle, Terry Ruas, and Bela Gipp. 2025. Voting or consensus? decision-making in multi-agent debate. *arXiv preprint arXiv:2502.19130*.

Alex Kim, Keonwoo Kim, and Sangwon Yoon. 2024. Debate: Devil's advocate-based assessment and text evaluation. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 1885–1897.

Geunwoo Kim, Pierre Baldi, and Stephen McAleer. 2023. Language models can solve computer tasks. *Advances in Neural Information Processing Systems*, 36:39648–39677.

Chaofan Li, Zheng Liu, Shitao Xiao, and Yingxia Shao. 2023a. Making large language models a better foundation for dense retrieval. *arXiv preprint arXiv:2312.15503*.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023b. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*.

Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Shuming Shi, and Zhaopeng Tu. 2024. Encouraging divergent thinking in large language models through multi-agent debate. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17889–17904.

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

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, and others. 2024a. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2023. Lost in the middle: How language models use long contexts. *arXiv preprint arXiv:2307.03172*.

Tongxuan Liu, Xingyu Wang, Weizhe Huang, Wenjiang Xu, Yuting Zeng, Lei Jiang, Hailong Yang, and Jing Li. 2024b. Groupdebate: Enhancing the efficiency of multi-agent debate using group discussion. *arXiv preprint arXiv:2409.14051*.

Yexiang Liu, Jie Cao, Zekun Li, Ran He, and Tieniu Tan. 2025. Breaking mental set to improve reasoning through diverse multi-agent debate. In *The Thirteenth International Conference on Learning Representations*.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, and others. 2023. Self-refine: Iterative refinement with self-feedback. *Advances in Neural Information Processing Systems*, 36:46534–46594.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Marvin Minsky. 1986. *Society of mind*. Simon and Schuster.

Jihwan Oh, Minchan Jeong, Jongwoo Ko, and Se-Young Yun. 2025. When debate fails: Bias reinforcement in large language models. In *Workshop on Reasoning and Planning for Large Language Models*.

OpenAI. 2025. Introducing gpt-5. Accessed: 2025-12-20.

Chau Pham, Boyi Liu, Yingxiang Yang, Zhengyu Chen, Tianyi Liu, Jianbo Yuan, Bryan A Plummer, Zhaoran Wang, and Hongxia Yang. 2024. Let models speak ciphers: Multiagent debate through embeddings. In *The Twelfth International Conference on Learning Representations*.

Stephen Robertson, Hugo Zaragoza, and others. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Eva Sharma, Chen Li, and Lu Wang. 2019. Bigpatent: A large-scale dataset for abstractive and coherent summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2204–2213.

Chenhui Shen, Liying Cheng, Ran Zhou, Lidong Bing, Yang You, and Luo Si. 2022. Mred: A meta-review dataset for structure-controllable text generation. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2521–2535.

Andries Smit, Nathan Grinsztajn, Paul Duckworth, Thomas D Barrett, and Arnu Pretorius. 2024. Should we be going mad? a look at multi-agent debate strategies for llms. In *Proceedings of the 41st International Conference on Machine Learning*, pages 45883–45905.

Saba Sturua, Isabelle Mohr, Mohammad Kalim Akram, Michael Günther, Bo Wang, Markus Krimmel, Feng Wang, Georgios Mastrapas, Andreas Koukounas, Nan Wang, and others. 2024. jina-embeddings-v3: Multilingual embeddings with task lora. *arXiv preprint arXiv:2409.10173*.

The State Grid Corporation of China. 2024. Introducing claude sonnet 4.5. Accessed: 2025-12-20.

The U.S. Department of Energy. 2023. Smart grid grants | department of energy. Accessed: 2025-12-20.

Qineng Wang, Zihao Wang, Ying Su, Hanghang Tong, and Yangqiu Song. 2024a. Rethinking the bounds of llm reasoning: Are multi-agent discussions the key? In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6106–6131.

Sijia Wang and Lifu Huang. 2024. Debate as optimization: Adaptive conformal prediction and diverse retrieval for event extraction. In *Proceedings of the conference Association for Computational Linguistics Meeting*, volume 1. Association for Computational Linguistics.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, and others. 2024b. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Andrea Wynn, Harsh Satija, and Gillian Hadfield. 2025. Talk isn't always cheap: Understanding failure modes in multi-agent debate. *arXiv preprint arXiv:2509.05396*.

Kai Xiong, Xiao Ding, Yixin Cao, Ting Liu, and Bing Qin. 2023. Examining inter-consistency of large language models collaboration: An in-depth analysis via debate. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and others. 2025a. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

An Yang, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoyan Huang, Jiandong Jiang, Jianhong Tu, Jianwei Zhang, Jingren Zhou, Junyang Lin, Kai Dang, Kexin Yang, Le Yu, Mei Li, Minmin Sun, Qin Zhu, Rui Men, Tao He, and 9 others. 2025b. Qwen2.5-1m technical report. *arXiv preprint arXiv:2501.15383*.

Yongjin Yang, Euiin Yi, Jongwoo Ko, Kimin Lee, Zhijing Jin, and Se-Young Yun. 2025c. Revisiting multi-agent debate as test-time scaling: A systematic study of conditional effectiveness. *arXiv preprint arXiv:2505.22960*.

Zhangyue Yin, Qiushi Sun, Cheng Chang, Qipeng Guo, Junqi Dai, Xuan-Jing Huang, and Xipeng Qiu. 2023. Exchange-of-thought: Enhancing large language model capabilities through cross-model communication. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 15135–15153.

Yang Yu, Lei Liu, Xiangyi Xu, and Shaoqian Bai. 2022. *Text detection methods, devices, computing equipment, and computer-readable storage media*, cn108829780b edition.

Aohan Zeng, Xin Lv, Qinkai Zheng, Zhenyu Hou, Bin Chen, Chengxing Xie, Cunxiang Wang, Da Yin, Hao Zeng, Jiajie Zhang, and others. 2025. Glm-4.5: Agentic, reasoning, and coding (arc) foundation models. *arXiv preprint arXiv:2508.06471*.

Daoze Zhang, Zhijian Bao, Sihang Du, Zhiyi Zhao, Kuangling Zhang, Dezheng Bao, and Yang Yang. 2025a. Re^2: A consistency-ensured dataset for full-stage peer review and multi-turn rebuttal discussions. *arXiv preprint arXiv:2505.07920*.

Marina Zhang, Owen Skipper Vallis, Aysegul Bumin, Tanay Vakharia, and Elie Bursztein. 2024. Retsim: Resilient and efficient text similarity. In *The Twelfth International Conference on Learning Representations*.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. 2025b. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*.

Zhenhai Zhang and Xiongyong Sun. 2012. *A method and system for automatically detecting academic misconduct literature*, cn101833579b edition.

Huaixiu Steven Zheng, Swaroop Mishra, Xinyun Chen, Heng-Tze Cheng, Ed H Chi, Quoc V Le, and Denny Zhou. 2023a. Take a step back: Evoking reasoning via abstraction in large language models. *arXiv preprint arXiv:2310.06117*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, and others. 2023b. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623.

A APPENDIX

A.1 Related work

A.1.1 Textual Duplication Detection

Textual duplication detection is a crucial computational linguistics task for identifying content replication across documents, playing a vital role in protecting academic integrity and intellectual property. Early methods primarily relied on lexical-level analysis, such as n-gram overlap quantification (Bensalem et al., 2014) and dynamic programming algorithms like Smith-Waterman for local alignment. Notably, Yu et al. (2022) introduced advaced syntactic-level processing—including sentence segmentation, lexical decomposition, and TF-IDF differential computation—for similarity matrix construction, as implemented in the Wanfang duplication detection platform.

While these character-matching methods achieve high precision in verbatim detection, they lack high-level semantic comprehension, making them susceptible to evasion through paraphrasing (e.g., synonym substitution and syntactic restructuring). Later advancements adopt distributed semantic representations, employing topic modeling (e.g., Latent Dirichlet Allocation) and word embeddings (e.g., Word2Vec (Mikolov et al., 2013)) for document similarity. Zhang and Sun (2012) proposed a hierarchical detection framework combining document-level keyword ranking with sentence-level synonymy detection, widely used in the CNKI duplication detection platform. Although these approaches capture surface-level semantic relationships, they struggle with complex semantic transformations. The rise of large language models (LLMs) has revolutionized reasoning tasks (Gu et al., 2024), yet their mechanisms inherently conflict with the reality that there is no strict partial order in project duplication detection.

A.1.2 Multi-Agent Debate Systems

Multi-Agent Debate (MAD) (Liang et al., 2024; Du et al., 2023; Xiong et al., 2023; Chen et al., 2024b) implements the "society of minds" framework through collaborative interactions among LLM agents. This approach addresses key limitations of single-model reasoning—such as confirmation bias, hallucinations, and logical inconsistencies—through iterative adversarial knowledge refinement. Empirical studies show that MAD improves reasoning via three mechanisms: (1) collective error correction, (2) perspective diversification, and (3) systematic reasoning reinforcement.

MAD has proven effective in diverse applications. In scientific discovery, Gottweis et al. (2025) uses tournament-style debates to generate and refine biomedical hypotheses. For legal judgment prediction, Chen et al. (2025a) combines MAD with reliability assessment to reduce reliance on large datasets. In event extraction, the Debate as Optimization (DAO) system (Wang and Huang, 2024) iteratively improves outputs without parameter tuning. However, MAD remains unexplored for duplication detection, a gap that our work bridges.

Recent advancements in MAD have primarily focused on three key dimensions:

(1) **Communication optimization**, where Pham et al. (2024) demonstrates the superiority of embedding-based interaction over natural language debate and Yin et al. (2023) further improves performance through refined optimization of communication mechanisms. Oh et al. (2025) proposed optimizing each debater's speech one-on-one to prevent error propagation;

(2) **Role specialization**, with Chan et al. (2023) and Kim et al. (2024) establishing that heterogeneous agent personas significantly outperform homogeneous configurations. Besides, Liu et al. (2025) achieves state-of-the-art MAD performance by employing diverse reasoning methods for each participating agent. Several studies show the important improvement of MAD through enhancing the heterogeneity of base models and model capabilities (Yang et al., 2025c; Wynn et al., 2025);

(3) **Decision-making efficiency**, where Liu et al. (2024b) introduces grouped debates to reduce computational overhead, while Kaesberg et al. (2025) systematically evaluates voting versus consensus protocols across task types. Fan et al. (2025) adaptively decides whether to initiate debate to reduce token consumption while improving accuracy.

However, these methodological refinements have predominantly targeted single-knowledge question answering scenarios, leaving their applicability to complex, multi-faceted tasks like project duplication detection largely unexplored.

Despite its strengths, MAD faces criticism. Studies show it outperforms single-agent reasoning mainly in zero-shot settings (Wang et al., 2024a), and debates may amplify biases when agents share training data (Estornell and Liu, 2024). For example, Wang et al. (2024a) finds that well-prompted single agents can match MAD with demonstrations, while Estornell and Liu (2024) shows debates often converge to majority opinions, reinforcing misconceptions. These limitations underscore the need for careful role diversity and consensus design—challenges our work tackles via tailored agent roles and task-specific voting. Our empirical results in project duplication detection further confirm MAD's superiority over single LLM.

A.2 Dataset details

Table 3: Dataset details of power scientific projects

Year	Number of projects	Average length (in words)	Average length (in tokens)
2022	223	24, 610	13,395
2023	292	26, 305	14,142
2024	318	29, 680	15,851
Total	**833**	**27, 140**	**14, 594**

Power Project Data The power project data used in this work consists of real projects from the State Grid Corporation of China. We anonymize sensitive information (e.g., applicant details) and randomly generating IDs, retaining only titles and text content. The original data of these projects is provided in the form of feasibility study reports, often existing in formats such as doc, docx and pdf. The collection, use, and processing of these data have been reviewed by ethics experts in the electricity field and permitted to be used for the research in this work. More information is shown in Table 3.

Cross-domain Data The patent data used in cross-domain experiments are from F-kind (Mechanical Engineering; Lightning; Heating; Weapons; Blasting) patents in bigPatent (Sharma et al., 2019). In details, we randomly sample 1,000 patents with their abstracts. Additionally, we rewrite 100 sampled patents as test cases with 5 different LLMs

(GLM-4.6 (Zeng et al., 2025), Qwen3-235B-A22B-Instruct-2507 (Yang et al., 2025a,b), GPT-5 (OpenAI, 2025), Gemini 3 pro (Deepmind, 2025) and Claude Sonnet 4.5 (Anthropic, 2025)) to simulate the duplicate detection scenarios commonly encountered in the real world. Cross-domain data are open-sourced in this repository.

Human Annotation We invite experts in power domain for power data annotation in task 1 and task 2 while provide compensation based on the average standard hourly wage according to working hours.

The annotation instructions are provided here: Annotation data consists of a batch of scientific and technological research projects in the field of electricity. Each project conducts some scientific and technological research related to the field of electricity. The data provides the project ID, name, and application content. The annotation task is based on human evaluation, selecting the top-5 most relevant projects from the preliminary shortlisted 30 related projects. Detection Criteria: There is duplication in research content, key technologies, the same scenario, or applications. Note, project data is protected under confidentiality agreements and is prohibited from dissemination.

A.3 Hyperparameter Analysis

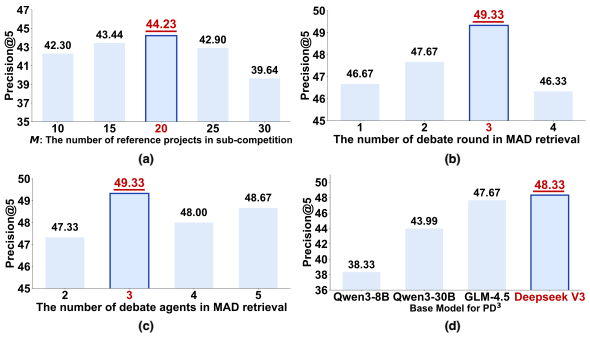

Figure 6: Hyperparameter analysis results. (a), (b), (c) and (d) presents the performance comparison across hyperparameters including: group size, number of debate rounds, debate agents and base models, respectively.

Hyperparameter Experiments Settings Building on prior findings regarding the parameter sensitivity of MAD(Smit et al., 2024), we're also interested in how sensitive is PD^3's performance to its hyperparameters. Therefore, we conduct systematic experiments on four core hyperparameters of the MAD framework: the number of candidate items M from preliminary retrieval, debate rounds, agent counts and agents' base models.

Hyperparameter Analysis (RQ4) As illustrated in Figure 6 (a), experimental results across varying values of M demonstrate a characteristic "peak-then-decline" trend in reasoning performance, with optimal performance achieved at $M = 20$. This phenomenon suggests that while increasing M enhances the agents' access to more global information, excessively large values introduce longer context, ultimately compromising reasoning quality.

To optimize computational costs, we conduct the following experiments on a 60-sample random subset of the original dataset. As shown in Figure 6 (b), when fixing the number of agents at 3, model performance improves with additional debate rounds but declines beyond 3 rounds. Figure 6 (c) reveals that the optimal performance is achieved with 3 agents and 3 debate rounds. These findings highlight two competing factors: (1) Sufficient debate rounds and agents facilitate multi-perspective analysis and comprehensive reasoning. (2) Excessive rounds or agents introduce diminishing returns due to error propagation and context overload, as discussed in (Estornell and Liu, 2024).

For base models, we additionally test PD^3 on Qwen3-8B, Qwen3-VL-30B-A3B-Instruct (Yang et al., 2025a) and GLM-4.5 (Zeng et al., 2025), for their representative characteristics at different parameter scales. Figure 6 (d) shows that PD^3 indeed demonstrates a certain scalability on different size LLMs, with performance gradually improving as the parameter size increases. At the same time, it also shows quite good performance on GLM-4.5, which is comparable to Deepseek V3 on common benchmarks, reflecting PD^3's certain deployment adaptability. See detailed hyperparameter analysis results in Table 9, Table 10 and Table 11.

While our PD^3 implementation in Table 1 doesn't adopt the empirically optimal configuration (3 agents + 3 rounds, compared to our 3+ 2 setting) due to resource constraints, we also evaluate PD^3 using the best hyperparameter configuration on the full dataset. The experimental results show that PD^3 further achieved 44.53% in the Prec@5 metric, representing an improvement of 0.3% (see in Table 5). Meanwhile, this also suggests that it is worthwhile to balance effectiveness and computational cost in practical applications.

Our comprehensive evaluation of all baseline methods mentioned in Section 3.2 on subsets reveals that PD^3 consistently maintains superior performance, even under the worst hyperparameter configurations. As evidenced by the quantita-

tive results in Table 6, this empirical observation demonstrates the framework's remarkable robustness against parameter variations. The sustained performance advantage suggests that PD^3's architectural design inherently compensates for non-ideal parameter settings, a characteristic particularly valuable for real-world deployment where optimal parameter tuning may not always be feasible.

A.4 Case Study and Runtime Performance

Case Study *Review Dingdang*'s workflow starts by retrieving 30 candidate reference projects based on vector distance. Following the round-robin competition format, the 30 candidate projects are divided into 15 parallel sub-competition tasks. Figure 5 details one such sub-competition: three expert agents first independently select their top-5 choices, then debate until the specified number of rounds is reached. In this case, Expert Agent A ultimately accepts the opinions of Expert Agents B and C for their reasonable opinions, and the senior expert makes the final decision. After completing all sub-competitions, a voting mechanism determines the final top-5 projects. In the feedback stage, specialized agents then produce both quantitative scores and qualitative assessments to the human expert.

Runtime Performance of *Review Dingdang*

To quantitatively assess the operational efficiency of the deployed system, we conduct performance evaluation focusing on three key metrics: (1) average execution time, (2) computational cost, and (3) token consumption during the detection process. Executed in parallel via Python's concurrent module using Volcengine's Model API (ByteDance, 2025), with input cost at $0.28 per million tokens and output cost at $1.14 per million tokens for Deepseek V3. Our results demonstrate an average execution time of **3 minutes** per project duplication detection. As detailed in Table 4, token consumption averages 4.42 million tokens per project (4.36 million prompt tokens/57.98 thousand output tokens) during debates, while the feedback stage requires 35.36 thousand tokens per project (31.75 thousand prompt tokens/3.61 thousand output tokens). Based on Volcengine's Model API pricing (ByteDance, 2025), it takes around **1.34 USD** per project. These metrics quantify the temporal and computational cost of PD^3 framework and *Review Dingdang* platform.

Table 4: Running time token consumption analysis.

Process Stage	Token Consumption Metrics	Average Token Usage input	output	total
Debate	per Debate	290k	4k	294k
	per Project	4, 358k	58k	4, 416k
Feedback	per Project	32k	3k	35k
Total	per Project	4, 390k	61k	4, 451k

A.5 Experiment settings details

General experiment settings We choose DeepSeek-V3-250324 as base-model for PD^3 and all LLM-associated methods. As an open-source model, it excels in both Chinese and English support and general benchmark performance (Wang et al., 2024b). In MAD round-robin retrieval, we set the number of agents and debate rounds (except initial round) to 3 and 2 respectively refer to the settings in (Du et al., 2023). Since we use the same, pre-randomly shuffled reference project information input in different baseline and hyperparameter experiments, we maintain consistent prompt settings and default base model hyperparameter settings (such as temperature, etc.), thus maintaining stability in multi-round experiments. In addition, referring to the settings of previous work (Chan et al., 2023; Fan et al., 2025; Liu et al., 2025) and insights from general LLM experiments settings (Liu et al., 2024a; Zeng et al., 2025), we report the one-round results in each experiment.

Experiment settings details on task 1 For the methods based on word frequency, vector distance retrieval, and LLM-as-a-Judge, we directly calculate the scores of the projects under detection and the preliminary retrieval results and then select the top-5 results with the highest scores. For DeepSeek V3 and DeepSeek R1, we directly input the project under detection with all preliminary retrieval results as prompts to generate outputs. For TourRank, We follow the original text setting, sequentially adopt a 30 $\rightarrow$ 20 $\rightarrow$ 10 $\rightarrow$ 5 $\rightarrow$ 2 filtering process, and run 10 times in parallel per case to determine the final result. For DMAD, we replace some of the original reasoning methods (limited to closed-ended questions) with Chain-of-Thought (Wei et al., 2022), Step-Back Prompting (Zheng et al., 2023a) and Self-Refine (Madaan et al., 2023; Kim et al., 2023), maintaining consistency in all other experimental settings.

Experiment settings details on task 2 For all

LLM-as-a-Judge methods (including LLM-as-a-Judge MAX, LLM-as-a-Judge AVG, and PD³'s LLM-as-a-Judge-based feedback with or without conclusion), we perform three independent generations and use the average value as the final score. While we employ the average of three scores as the final evaluation metric, instances may occur where the LLM assigns identical scores to both $P_{\text{u-A}}$ and $P_{\text{u-B}}$ in the test set. For fairness, our scoring protocol differs between evaluation settings in such cases: (1) Under the "Origin Group" setting, these cases receive a score of 0. (2) Under the "Weighted Group" setting, we assign the score corresponding to the less frequent label in the annotation (e.g., a 2:1 ratio would yield 0.33 points). This is based on our findings in the expert manual annotation: From multiple evaluation perspectives, it is difficult to make confident decision among several randomly selected comparison cases. We believe that this setting is closer to the granularity of comparison methods and manual detection.

A.6 Experiment results details

Experiment on task 1 results details For vector distance-based methods, we conduct additional experiments beyond those reported in the main text. Due to space limitations, Table 1 presents only the top-performing method from each category, while Table 5 in the appendix provides complete experimental results. Within each category, the bold entries indicate the methods selected for Table 1 based on superior performance. Table 1 also presents the performance of PD^3 under the best hyperparameters configuration (3 agents + 3 debate rounds).

Table 6 further shows the performance of all methods on small-scale datasets used for hyperparameter analysis. As described in section A.3. Combining the results of hyperparameter analysis, PD^3 outperforms other baseline methods under different hyperparameter settings, further indicating the robustness of the method.

Experiment on ablation results details Table 7 shows the detailed results of ablation analysis.

Experiment on cross-domain dataset results details Table 8 shows the detailed experimental results of cross-domain analysis.

Experiment on hyperparameters results details Table 9, Table 10, and Table 11 respectively show the detailed experimental results of the analysis for four key hyperparameters in PD^3 - the number of candidate items M from preliminary retrieval, debate rounds, and agent count.

A.7 Potential Risks

We acknowledge that there are certain application risks in the PD^3 framework. Focused on more efficiently and accurately identifying duplication, while providing informative feedback, the design of the PD^3 framework centers on the review process and collaboration among multiple agents. For practical applications, adding more security protection measures to prevent malicious attacks is necessary, for example, adding an additional large model discrimination layer to prevent prompt injection attacks, and adding additional cybersecurity methods to prevent attacks such as DDoS. The transparent and clear design makes PD^3 easy to integrate with any security strategy. Due to the setting of the core scope and targets of this work, we do not discuss in details security risks in this paper.

Table 5: Additional experiment result of task 1

Method	Prec@5	Match@K				
		K = 1 (Hit Rate@5)	K = 2	K = 3	K = 4	K = 5
RETSim	27.73	13 | 0.0393	2 | 0.0060	0 | 0.0000	0 | 0.0000	0 | 0.0000
gte-7B as reranker+ Task INST	37.70	293 | 88.52	210 | 63.44	93 | 28.10	26 | 7.85	**2 | 0.60**
gte-7B as reranker(w/o INST)	**38.97**	**298 | 90.03**	**212 | 64.05**	**105 | 31.72**	**29 | 8.76**	1 | 0.30
gte-1.5B+ EN default INST	36.92	290 | 87.61	209 | 63.14	92 | 27.79	19 | 5.74	1 | 0.30
gte-1.5B+ CN default INST	37.22	295 | 89.12	207 | 62.54	94 | 28.40	19 | 5.74	1 | 0.30
gte-1.5B with instruction (+ Task INST)	**37.82**	**294 | 88.82**	**213 | 64.35**	**96 | 29.00**	**21 | 6.36**	**2 | 0.60**
bge as reranker	24.35	260 | 78.55	113 | 34.13	25 | 7.55	**4 | 1.21**	1 | 0.30
jina as reranker	**27.98**	**275 | 83.08**	**141 | 42.60**	**42 | 12.69**	1 | 0.30	**1 | 0.30**
PD^3 (Best Hyperparameters)	**44.53**	**313 | 94.56**	**238 | 71.90**	**133 | 40.18**	**47 | 14.20**	**6 | 1.81**

Table 6: Additional experiment result on subset of task 1

Method	Prec@5
Random	18.00
ROUGE-L	23.00
BM25	33.00
gte-1.5B	37.33
gte-1.5B with instruction	39.33
gte-7B as reranker	39.67
Qwen3-8B as reranker	35.00
jina as reranker	28.33
gte-7B as reranker+ Task INST	40.33
gte-7B as reranker(w/o INST)	39.67
gte-1.5B+ EN default INST	37.33
gte-1.5B+ CN default INST	39.67
gte-1.5B with instruction (+ Task INST)	39.33
bge as reranker	26.67
jina as reranker	28.33
DeepSeek V3	37.00
DeepSeek R1	42.00
LLM-as-a-Judge	40.67
DeepSeek V3 Voting	45.67
MAD Vanilla	43.00
DMAD	41.00
MAD Traversal	43.33
MAD Random	45.33
MAD Sliding Window	<u>46.00</u>
PD^3 (Worst hyperparameter)	**46.33**
PD^3 (Ours)	**47.67**
PD^3 (Optimal hyperparameter)	**49.33**

Table 7: Ablation experiment result

Method	Prec@5	Match@K				
		K = 1 (Hit Rate@5)	K = 2	K = 3	K = 4	K = 5
gte-1.5B **(VD)** (w/o MAD, w/o round-robin)	38.13	296 | 89.43	219 | 66.16	97 | 29.31	18 | 5.44	1 | 0.30
DeepSeek V3 Voting **(LLM with Voting) (VD)** (w/o MAD, with round-robin)	42.05	306 | 92.45	224 | 67.67	122 | 36.86	38 | 11.48	<u>6 | 1.81</u>
MAD Vanilla **(MAD Vanilla)** (with MAD, w/o round-robin)	39.64	307 | 92.75	229 | 69.18	<u>129 | 38.97</u>	**42 | 12.69**	3 | 0.91
MAD Traversal **(Traversal)** (with MAD, w/o round-robin)	42.11	304 | 91.84	224 | 67.27	125 | 37.76	40 | 12.08	4 | 1.21
MAD Random **(Random)** (with MAD, w/o round-robin)	42.72	<u>309 | 93.35</u>	<u>234 | 70.69</u>	125 | 37.76	35 | 10.57	4 | 1.21
MAD Sliding Window **(Sliding)** (with MAD, w/o round-robin)	<u>43.44</u>	307 | 92.75	233 | 70.39	**133 | 40.18**	**42 | 12.69**	4 | 1.21
PD^3 (ours) (with MAD, with Round-robin)	**44.23**	**310 | 93.66**	**238 | 71.90**	**133 | 40.18**	<u>41 | 12.39</u>	**10 | 3.02**

Table 8: Cross-domain experiments result.

Method	Prec@5	Match@K				
		K = 1 (Hit Rate@5)	K = 2	K = 3	K = 4	K = 5
Random **(Random)**	18.80	67 | 67.00	25 | 25.00	2 | 2.00	0 | 0.00	0 | 0.00
ROUGE-L **(WF)**	43.60	92 | 92.00	72 | 72.00	39 | 39.00	13 | 13.00	2 | 2.00
BM25 **(WF)**	44.40	91 | 91.00	73 | 73.00	41 | 41.00	15 | 15.00	2 | 2.00
gte-1.5B **(VD)**	52.00	94 | 94.00	77 | 77.00	56 | 56.00	28 | 28.00	5 | 5.00
gte-1.5B with instruction **(VD)**	47.20	92 | 92.00	73 | 73.00	48 | 48.00	21 | 21.00	2 | 2.00
gte-7B as reranker **(VD)**	22.60	74 | 74.00	30 | 30.00	8 | 8.00	1 | 1.00	0 | 0.00
Qwen3-8B as reranker **(VD)**	43.80	92 | 92.00	68 | 68.00	46 | 46.00	11 | 11.00	2 | 2.00
jina as reranker **(VD)**	19.40	57 | 57.00	26 | 26.00	11 | 11.00	3 | 3.00	0 | 0.00
DeepSeek V3 **(LLM)**	76.00	**100 | 100.00**	98 | 98.00	91 | 91.00	68 | 68.00	23 | 23.00
DeepSeek R1 **(LLM)**	80.00	**100 | 100.00**	98 | 98.00	97 | 97.00	79 | 79.00	26 | 26.00
LLM-as-a-Judge **(LLM)**	80.60	**100 | 100.00**	**99 | 99.00**	97 | 97.00	80 | 80.00	27 | 27.00
TourRank **(LLM)**	76.80	**100 | 100.00**	**99 | 99.00**	91 | 91.00	72 | 72.00	22 | 22.00
MAD Vanilla **(MAD)**	79.60	**100 | 100.00**	**99 | 99.00**	93 | 93.00	75 | 75.00	31 | 31.00
DMAD **(MAD)**	71.20	**100 | 100.00**	98 | 98.00	86 | 86.00	57 | 57.00	15 | 15.00
PD^3 MAD Round-robin (Ours)	**83.60**	**100 | 100.00**	**99 | 99.00**	**98 | 98.00**	**84 | 84.00**	**37 | 37.00**

Table 9: Hyperparameter analysis experiment result on the group size.

Round-robin initial item count	Prec@5	Match @ K				
		K = 1 (Hit Rate@5)	K = 2	K = 3	K = 4	K = 5
10	42.30	307 | 92.75	229 | 69.18	119 | 35.95	39 | 11.78	6 | 1.81
15	43.44	309 | 93.35	229 | 69.18	132 | 39.88	**45 | 13.60**	4 | 0121
20 (ours)	**44.23**	**310 | 93.66**	**238 | 71.90**	**133 | 40.18**	41 | 1239	**10 | 3.02**
25	42.90	307 | 92.75	229 | 69.18	129 | 38.97	42 | 12.69	3 | 0.91
30	39.64	307 | 92.75	229 | 69.18	129 | 38.97	42 | 12.69	3 | 0.91

Table 10: Hyperparameter analysis experiment result on the debate round.

Debate rounds	Prec@5	Match@K				
		K = 1 (Hit Rate@5)	K = 2	K = 3	K = 4	K = 5
1	46.67	56 | 93.33	45 | 75.00	28 | 46.67	9 | 15.00	**2 | 3.33**
2	47.67	57 | 95.00	**47 | 78.33**	29 | 48.33	8 | 13.33	**2 | 3.33**
3	**49.33**	**58 | 96.67**	45 | 75.00	**33 | 55.00**	**11 | 18.33**	1 | 1.67
4	46.33	57 | 95.00	43 | 71.61	30 | 50.00	8 | 13.33	1 | 1.67

Table 11: Hyperparameter analysis experiment result on the number of debate agent.

Agent Count	Prec@5	Match@K				
		K = 1 (Hit Rate@5)	K = 2	K = 3	K = 4	K = 5
2	47.33	56 | 93.33	**48 | 80.00**	25 | 41.67	**12 | 20.00**	**1 | 1.67**
3	**49.33**	**58 | 96.67**	45 | 75.00	33 | 55.00	11 | 18.33	**1 | 1.67**
4	48.00	57 | 95.00	46 | 76.67	30 | 50.00	10 | 16.67	**1 | 1.67**
5	48.67	56 | 93.33	45 | 75.00	**34 | 56.67**	10 | 16.67	**1 | 1.67**

A.8 Prompt Templates

Prompt template for MAD round-robin retrieval – initial round

You are an expert in the field of power conducting project duplication detection named {expert_name}.
Please select the five most relevant projects to the project under detection based on the following detection criteria and project information.
As an independent expert, you have no preconceived biases towards the research content of each project and focus solely on determining the best choice.
Detection Objective:
Based on predefined detection criteria and discussion procedures, strictly discuss and determine five candidate projects that are most relevant to the project under detection.
Detection Criteria:
{detection_criteria}
Project Under Detection Information:
{project_under_detection_info}
Candidate Relevant Reference Project Information:
{candidates_project_info}
Please select the five most relevant projects you believe and briefly explain the reasons for your selection in a complete sentence.

Prompt template for MAD round-robin retrieval – debate round

You are an expert in the field of power conducting project duplication detection named {expert_name}.
You are participating in a project duplication detection debate involving fellow experts, aiming to select the five most relevant candidate projects to the project under detection from several options. Currently, it is the {round_num}th round of debate.
I will provide you with the detection objectives, detection criteria, relevant project information, and records of previous rounds of debate.
Please make a statement based on the previous discussions. You can: respond to the opinions of other experts, present new arguments to support your choices; or adopt the opinions of other experts, modify your previous choices and explain the reasons; or also question the choices and statements of other experts.
Detection Objective:
Based on predefined detection criteria and discussion procedures, strictly discuss and determine five candidate projects that are most relevant to the project under detection.
Detection Criteria:
{detection_criteria}
Project Under Detection Information:
{project_under_detection_info}
Candidate Relevant Reference Project Information:
{candidates_project_info}
Records of previous debate: {debate_records}
Please make a brief statement in a complete sentence to express your views.

Prompt template for MAD round-robin retrieval – senior expert

As a senior expert in the field of power for project duplication detection, you are responsible for organizing expert debate to select the five most relevant projects among several candidate related projects.
You will serve as a discussion reviewer in this debate, evaluating the experts' debate and determining the final five selected projects.
Detection Objective:
Based on predefined detection criteria and discussion procedures, strictly discuss and determine five candidate projects that are most relevant to the project under detection.
Detection Criteria:
{detection_criteria}
Project Under Detection Information:
{project_under_detection_info}
Candidate Relevant Reference Project Information:
{candidates_project_info}
Records of debate:
{debate_records}
Please analyze the experts' consensus, and finally, output the list of project numbers in order of relevance after [RESULT].

Prompt template for LLM-as-judge-based feedback – duplication scoring

You are an expert in the field of power conducting project duplication detection.
For a given project under detection, you are provided with the five most relevant historical projects from the provided database, as well as the review experts' conclusion on the relevant content of the reference projects and the project under detection.
Your task is to score the degree of duplication of the project under detection according to the detection criteria.
Detection Criteria:
Scoring is on a 10-point scale: 1 is the lowest, indicating that all historical projects are basically unrelated to the project under detection; 4-6 is in the middle, indicating that multiple projects from the historical projects have duplication with the project under detection in some dimension; 10 is the highest, indicating that one or more reference projects are completely identical to the project under detection. Encourage scores with differentiation.
When scoring, it is necessary to comprehensively consider the similarity of the candidate relevant projects in terms of research themes, core technologies, and application scenarios.
It should be noted that if one of the five reference projects is highly relevant to the project under detection, a higher score should be given. Only when all five reference projects are not sufficiently relevant should a lower score be given.
It should be noted that the five reference projects provided may not be highly relevant to the project under detection.
Project Under Detection Information:
{project_under_detection_info}
Candidate Relevant Reference Project Information:
{candidates_project_info}
Conclusion of detection Experts:
{expert_conclusion}
You need to provide the analysis reasons first, and then give the score in the form of '[RESULT]score', where score is an integer from 1 to 10.

Prompt template for rewriting bigPatent in cross-doamin experiments

You are a staff member of a patent office, and your task is to write a new patent abstract based on the given original patent.
The requirement is to rewrite it based on the original patent, replacing key technologies, polishing the target issue, or changing the application scenario, so that the new patent is different from the original patent and can avoid being detected as a duplicate. You are encouraged to make substantial text modifications.
Please directly output the new patent abstract; do not include any other content or markings.
Original Patent Abstract: {origin_patent}