Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

SupersedeRank: Query-Time-Aware Reranking of Superseded CVSS Evidence in Retrieval-Augmented Generation

Doygun Demirol

ddemirol@bingol.edu.tr

Bingöl University

Research Article

Keywords: Retrieval-augmented generation, Temporal information retrieval, Query-time validity, Temporal reranking, Vulnerability management

Posted Date: November 18th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11160774/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

SupersedeRank: Query-Time-Aware Reranking of Superseded CVSS Evidence in Retrieval-Augmented Generation

Doygun Demirol[1*]

[1*]Bingöl University, Bingöl, Türkiye.

Corresponding author(s). E-mail(s): ddemirol@bingol.edu.tr;

Abstract

Retrieval-augmented generation systems ground their answers in evidence retrieved from a document corpus. Over time, older records may remain semantically relevant even after being superseded by newer records. This phenomenon can be systematically examined in vulnerability management because the National Vulnerability Database (NVD) records the old and new values of revised CVSS assessments and the corresponding change dates. In this study, we develop SupersedeRank, a query-time-aware, explainable temporal reranking method with no learned parameters. The method preserves first-stage retrieval candidates. Evidence in the query-targeted source chain is moved behind protected candidates only if it is inferred to have been superseded by a newer and different assessment visible at query time, without coefficients or thresholds. Reference labels were programmatically derived from explicit old-to-new value transitions in official change records without additional human annotation. In a future-entity temporal test comprising 1,506 queries for 1,356 vulnerabilities unseen during training, SupersedeRank-Rule-Target ranked current evidence first for 99.8% of queries. It preserved current evidence for every query and left no superseded target evidence among the top-ten candidates. The strongest baseline, BM25 with a recency bonus, promoted current evidence at a comparable rate but retained superseded evidence among the top ten. The experimental scope is limited to atomic CVSS replacement events within the same source chain and change event.

Keywords: Retrieval-augmented generation, Temporal information retrieval, Query-time validity, Temporal reranking, Vulnerability management

1 Introduction

Retrieval-Augmented Generation (RAG) [22] aims to overcome the knowledge cutoff inherent in a language model's training data by grounding its responses in evidence retrieved from an external document repository. However, versioned knowledge bases and official record systems accumulate records about the same entity over time, and records that were accurate when created may coexist with newer records that supersede them [34]. Retrieval and reranking components based solely on semantic relevance cannot distinguish between these two types of records and may therefore rank superseded records highly and mislead the model's response. This temporal information retrieval problem requires ranking evidence not only by its relevance to the query but also by its validity at query time. Vulnerability management is a domain where this problem can be examined in a traceable manner. For each vulnerability identified by a Common Vulnerabilities and Exposures (CVE) identifier, the NVD publishes a Common Vulnerability Scoring System (CVSS) assessment comprising a metric vector of exploitation conditions, a base score, and a severity rating [13, 14]. Organizations other than the NVD that report the same vulnerability can also publish their own assessments and revise them over time. The NVD's change history stores the values removed and added during each revision, along with date and source information. This record structure clearly demonstrates that the validity of an assessment is not a static attribute of the record but depends on the relationship between the record and query time. Figure 1 illustrates a representative atomic replacement within a single source chain, in which a CVSS 3.1 assessment with base score 9.8 (Critical) is replaced by one with base score 7.3 (High). The initial assessment is current at a query time before the event but superseded at a query time after it, when the new assessment becomes current.

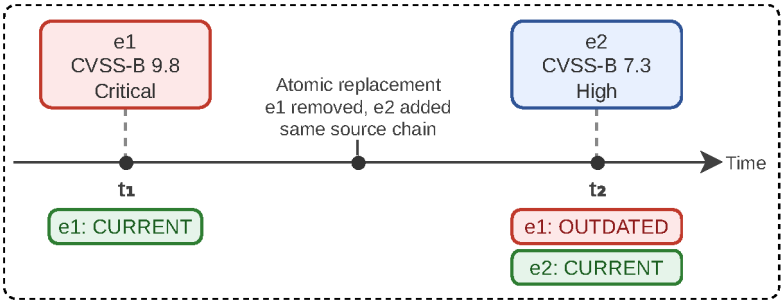


Fig. 1 Query-time-dependent CVSS validity.

Retrieval and reranking methods based solely on semantic relevance fail to detect this relationship because the current and superseded records share the same vulnerability, source, and CVSS version. While a recency bonus based on document age partially addresses the issue by promoting the newer record, it does not remove the superseded record from the context. Furthermore, its effectiveness depends on the temporal proximity of the query to the replacement event. SupersedeRank addresses

these limitations through a source-chain-aware evidence suppression policy integrated into the existing retrieval pipeline. The method preserves the first-stage candidates and moves a candidate behind the protected candidates only if a newer and different assessment is visible at query time within the source chain targeted by the query. It requires no coefficients, thresholds, or training. Figure 2 illustrates this behavior by comparing the same candidate list across the methods. BM25 cannot distinguish between the current and superseded records. The recency bonus promotes the current record but leaves the superseded record among the top-ten candidates. In contrast, SupersedeRank preserves current assessments from other sources while moving only the superseded target record outside the top ten.

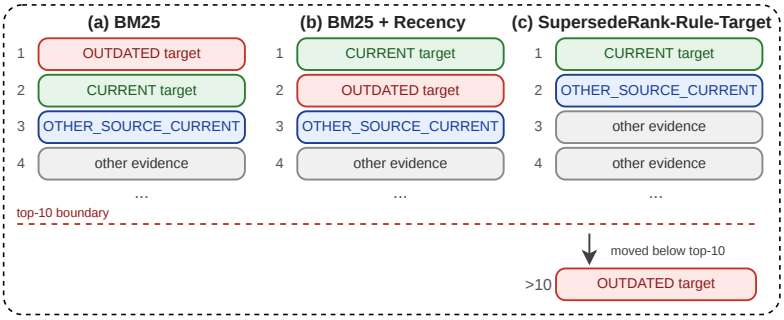


Fig. 2 Comparison of reranking behaviors.

Experimental verification was deliberately limited to atomic CVSS replacement events within the same source chain, in which the old value is removed, a new value is added, and the metric vector changes. Although descriptions, product lists, and patch information may also become superseded, distinguishing substantive updates from rewording requires human validation; restricting the scope to atomic events therefore ensures that the labels reflect the measured concept. The research gap arises from the fact that CVSS revisions explicitly recorded in the NVD change history have not been addressed as a query-time-dependent reranking problem (Section 2). This gap leads to four research questions: (RQ1) whether existing methods can distinguish between current and superseded evidence within the same source chain, (RQ2) whether SupersedeRank can satisfy both objectives simultaneously, (RQ3) what trade-off results from the scope of suppression, and (RQ4) whether the results hold for previously unseen vulnerabilities and across different query-time distances. These questions are answered in Section 6. The contributions of this study can be summarized as follows. The first is a temporal reranking problem that defines evidence validity based on the source chain and query time. The second is a temporal CVSS assessment dataset featuring reference labels derived from official CVSS replacement records without additional human annotation. The third is SupersedeRank-Rule-Target, a parameter-free and explainable method that suppresses superseded evidence in the target chain while preserving current evidence. The fourth consists of a future-entity evaluation protocol based on CVEs unseen during training, together with results supported by CVE-level

cluster bootstrap. These contributions are limited to retrieval and reranking quality; the accuracy of the generated answers was not evaluated.

2 Related Work

The problem outlined in the introduction stems from the fact that retrieval and reranking studies in RAG systems typically focus on semantic relevance. While these studies measure the relevance of evidence to a query, they do not separately evaluate its validity at query time. This distinction becomes important when information changes over time because superseded records may retain their semantic relevance. Research on temporal information retrieval, questions whose answers change over time, knowledge conflicts, and RAG applications in cybersecurity addresses different aspects of this distinction and collectively forms the background of the problem examined in this study.

2.1 Retrieval, reranking, and temporal information

RAG [22] retrieves candidate evidence from a document pool and re-ranks it before presenting the top-ranked items to the generation model as context. First-stage retrieval relies on lexical matching such as BM25 [31] or on dense retrievers, whose result lists can be merged with Reciprocal Rank Fusion [6]; second-stage rerankers such as cross-encoders [30] and learning-to-rank models [25] then refine the ranking. A common shortcoming of these methods is that they measure the relevance of the evidence to the question but do not account for its validity at query time. SupersedeRank introduces a block-ranking component to this process that relies solely on validity judgments, independent of relevance scores. The role of time in information retrieval is examined in two comprehensive surveys [2, 18]. While document-time-based priors [24] and recency ranking [9] treat recent documents as more valuable, Vu et al. [32] emphasize the need to explicitly manage freshness in the context of rapidly changing facts. For time-dependent questions, SituatedQA [36] incorporates temporal context into the query. Additionally, Dhingra et al. [8] examine facts conditioned on timestamps, while TemporalWiki [16] tracks changing facts across version histories. RealTimeQA [19] and PAT-Questions [27] regularly update questions with time-varying answers; meanwhile, ChronoQA [4] provides a time-sensitive RAG dataset, and Wallat et al. [33] measure the temporal blind spots of language models. A common assumption in these studies is that freshness equates to validity, or that validity can be defined at the answer level. However, this assumption fails in the context of security records. An old assessment that has never been revised may remain valid, whereas a newer and different assessment from the same source may supersede it. Therefore, SupersedeRank models not document age, but rather the visibility at query time of a newer and different assessment within the same source chain.

2.2 Superseded evidence, knowledge conflicts, and vulnerability intelligence

The impact of superseded information on RAG has been most directly measured by HoH [28], which generates pairs of current and superseded evidence from Wikipedia version histories and is the dataset most closely related to this study. NuggetIndex [35] stores atomic information units as records managed by validity intervals and lifecycle states. Behaviors exhibited when information in the model's memory conflicts with evidence in the context have been investigated through entity-swapping experiments by Longpre et al. [26] and under multiple conflicting passages by Chen et al. [3]. Furthermore, Xu et al. [34] surveys this literature, Kortukov et al. [20] measures conflict behavior using real-world documents, and Zhou et al. [37] reconciles conflicts during the decoding stage. Regarding source attribution, Rashkin et al. [29] defines a framework for measuring the attribution of generated text to evidence. Existing studies typically define validity at the document level, mostly through human annotation or indirectly from version differences, and address conflicts during the generation stage. However, which evidence supersedes which and when this occurs are not modeled during retrieval. In contrast, this study derives labels directly from official change records and applies the validity relationship before generation, specifically during the reranking stage.

The quality of vulnerability records and their evolution over time have been examined from various perspectives in the literature. Dong et al. [10] document inconsistencies among vulnerability reports; Anwar et al. [1] address errors in NVD records; Croft et al. [7] highlight issues regarding the timeliness of datasets; and Li and Paxson [23] analyze the behavior of security patches over time. Regarding CVSS specifically, Johnson et al. [17] measure inter-evaluator inconsistencies within a Bayesian framework, while Elbaz et al. [11] and Kühn et al. [21] predict CVSS vectors at the time of disclosure, and EPSS [15] estimates the probability of exploitation. Among language model-based systems, ProveRAG [12] aims to make the sources of answers traceable, whereas ChatNVD [5] focuses on question-answering based on NVD records. Although these studies generally address record quality, score prediction, or answer generation, they do not utilize the temporal revision history of records as a validity signal during retrieval.

When the relevant literature is considered collectively, three fundamental shortcomings emerge. Studies on temporal retrieval conflate recency with validity, research on superseded evidence defines validity labels mostly indirectly and at the document level, and cybersecurity RAG studies do not use official revision records as a validity signal during retrieval. None of the reviewed studies presents an approach that treats CVSS revisions in the NVD change history as a reranking problem dependent on the source chain and query time and evaluates it on vulnerabilities unseen during training. SupersedeRank addresses this gap by defining validity through the relationship between evidence and query time, constructing a dataset from official records, and applying a rule based only on fields observable by query time. This measurable problem definition is detailed in the following section.

3 Problem Definition

To measure the problem introduced above, the concepts of evidence, source chain, CVSS version, query time, query-time-visible evidence pool, and validity must be defined together. Otherwise, the labels become ambiguous or depend on information available after query time. The definition is based on the premise that validity is not a static property of evidence but depends on the relationship between evidence and query time. Since assessments from different sources do not automatically supersede one another, validity can only be defined within the same source chain. Therefore, the distinction between source chains is an essential part of the definition.

3.1 Temporal evidence and query time

An item of evidence represents a single CVSS assessment that is active for a specific vulnerability, source, and period, and is expressed by the tuple

$$e = (c_e,\ s_e,\ \nu_e,\ \mathbf{v}_e,\ b_e,\ g_e,\ I_e,\ \gamma_e) \tag{1}$$

where c_e denotes the CVE ID, s_e the normalized assessment source (`source_key`), ν_e the CVSS version, $\mathbf{v}_e$ the metric vector, b_e the base score, g_e the severity rating, $I_e = [t_e^-,\, t_e^+)$ the observed active interval, and $\gamma_e \in \{0, 1\}$ the left-censoring flag. The interval start t_e^- indicates when the assessment was first observed in the change history (`observed_from`), while $t_e^+ = \infty$ indicates that the assessment remains active at the end of the observation period. If $\gamma_e = 1$, the assessment became active before the observation window. In this case, t_e^- represents only the lower bound of the observation window rather than the actual publication or activation time.

The primary query is source-conditioned and is represented by the tuple

$$q = (c_q,\ s_q,\ \nu_q,\ t_q) \tag{2}$$

The query requests the CVSS assessment at version ν_q that is valid at query time t_q for vulnerability c_q from source s_q. Queries that do not specify a source ($q^{\text{agn}} = (c_q, t_q)$) are excluded from the experimental scope because multiple assessments may be simultaneously valid. The query-time-visible evidence pool consists of all evidence observed by time t_q within the observation window and is defined as

$$\mathcal{D}(t_q) = \{\, e \ :\ t_e^- \le t_q \,\} \tag{3}$$

This pool is not limited to evidence concerning the vulnerability specified in the query. The retrieval component selects up to N candidates from this pool, denoted by $\mathcal{R}_N(q) \subseteq \mathcal{D}(t_q)$, and assigns a relevance score $r(e, q)$ to each candidate. Evidence observed after query time is not included in the pool. This rule applies consistently across retrieval, reranking, and labeling.

3.2 Source-conditioned evidence chains

Multiple sources may simultaneously publish their own assessments for the same CVE, and a single source may maintain separate assessments for different CVSS versions. Consequently, evidence is organized into source-conditioned chains defined by the key

$$\kappa(e) = (c_e,\ s_e,\ \nu_e) \tag{4}$$

A chain represents the sequence of successive assessments issued by a specific source for a vulnerability under a single CVSS version. A newer assessment within the same chain may supersede the previous assessment, whereas assessments from different chains do not automatically supersede one another. An assessment is not considered superseded merely because another source subsequently publishes a different value. Similarly, a CVSS 3.1 assessment from a given source is not superseded merely because the same source publishes a CVSS 4.0 assessment. The target chain of a query, $\kappa(q) = (c_q, s_q, \nu_q)$, is determined directly from the query fields and does not require any label information.

3.3 Current and superseded evidence

The primary evaluation labels are binary and are defined only for evidence within the query's target chain. Given a piece of evidence e from Equation (1) and a query q from Equation (2), with $\kappa(e) = \kappa(q)$, the labels are defined as follows:

$$\ell(e,q) = \begin{cases} \text{CURRENT}, & t_e^- \le t_q < t_e^+, \\ \text{OUTDATED}, & t_e^+ \le t_q \text{ and } I_e \text{ ends with an atomic replacement} \\ & \text{in the same chain} \end{cases} \tag{5}$$

The labels are not permanently assigned to evidence but vary according to the relationship between the evidence and query time. The same piece of evidence receives the CURRENT label at a query time preceding the replacement event and the OUTDATED label at a query time following it. The OUTDATED label is based exclusively on atomic replacements occurring within the same chain and the same change event. The addition or removal of an assessment alone, the removal and re-addition of the same value, or the matching of removal and addition records across different events does not produce an OUTDATED label. Records for which the chain or active interval cannot be reliably established are marked as `EXCLUDED_AMBIGUOUS` and excluded from query generation. Evidence pertaining to the same vulnerability but belonging to a different source chain is not labeled according to Equation (5). Evidence that is active at query time is instead marked with the metadata `OTHER_SOURCE_CURRENT`. This metadata does not constitute a third class and is used only for graded relevance and error analysis. Such evidence is not considered superseded and is not included in the superseded evidence rate. Left-censored assessments are retained with $\gamma_e = 1$ and t_e^- set to the lower bound of the observation window, which serves only as a lower bound for age; the OUTDATED label always requires a subsequent explicit replacement event within the window.

3.4 Ranking objective and evaluation condition

The reference relevant set for the main query consists of evidence belonging to the target chain and valid at query time, expressed as

$$\mathcal{G}(q) = \{e \in \mathcal{D}(t_q) : \kappa(e) = \kappa(q) \text{ and } \ell(e, q) = \text{CURRENT}\}. \tag{6}$$

With this definition, retrieval and ranking errors can be measured separately by classifying evidence from different vulnerabilities as distractors, superseded assessments within the same chain as temporal hard negatives, and active assessments from other sources as non-target but active evidence. The reranker reorders the set $\mathcal{R}_N(q)$ returned by the retrieval component without adding new candidates or removing existing candidates. Candidates are evaluated according to the combined criterion

$$\sigma(e, q) = f\big(r(e, q), p(e, q)\big), \tag{7}$$

where $p(e, q)$ represents a judgment regarding the candidate's validity at query time. The function f is a ranking rule that combines semantic relevance with query-time validity. In this study, it is implemented as a parameter-free block ranking (Section 5). The objective is to preserve the current evidence in $\mathcal{G}(q)$ within the top k candidates while suppressing target-chain evidence labeled as OUTDATED by moving it outside the top k positions. In this study, k is set to 10, and these two objectives are evaluated using separate metrics (Section 6). The value of $p(e, q)$ is calculated solely from information available by time t_q. Label information, replacement relationships, and revisions occurring after t_q are not included in the reranker's input.

4 Dataset and Temporal Split

The reliability of the labels described in the Problem Definition section depends on the accurate processing of the NVD change history. Consequently, the data generation process entails more than mere record collection; it involves parsing each change event, establishing source-conditioned chains, replaying a timeline for each chain, and generating labels only from changes that can be reliably interpreted. All statistics regarding the query-time-dependent evaluation dataset generated in this study are automatically derived from the manifest and profile files of the production pipeline. The NVD's CVE API, which forms the basis of the dataset, provides the current record and available CVSS assessments for each vulnerability, while the change history API provides all changes made to each record over time, together with their old and new values. Within the observation window from January 1, 2024, to September 14, 2026, 156,270 CVE records and 1,590,753 raw change events were downloaded. These change events also include older vulnerabilities updated during this period.

4.1 Source-conditioned timeline

An activity interval is generated for each assessment by chronologically replaying supported CVSS addition and removal events from the change history within source-conditioned chains (Equation (4)). The source identifier is extracted from the source

field of the event and normalized using an alias mapping. Since record formats changed over time, source traceability is established through this field rather than through the vector text. Assessments that became active before the observation window are marked as left-censored. Events in which the same value was removed and re-added are not counted as replacement events and are retained as negative controls. The 131 cross-event candidates identified by matching removal and addition records from different events require additional assumptions. Therefore, they are excluded from the main dataset and retained only for sensitivity analysis. The 94 chains whose states cannot be reliably replayed are marked as ambiguous and excluded from the main dataset. Only two of these chains affect an atomic replacement event.

4.2 Replacement events and query-time labeling

For an event to be classified as an atomic replacement event, it must contain exactly one removed value and one added value for the same CVE and CVSS version within the same change event, both vectors must be parseable, at least one metric must have changed, and the event must result from a genuine reassessment rather than an administrative action. Records that do not meet these conditions are excluded from the dataset with the reason for exclusion recorded. The transition from the raw change history to a reliable and temporally distinct dataset is performed through a sequence of filters. Figure 3 summarizes this pipeline and shows the records excluded at each stage in separate branches. The final split boxes in the figure indicate the Base-query-eligible subset, while the event counts before filtering are presented in Section 4.4.

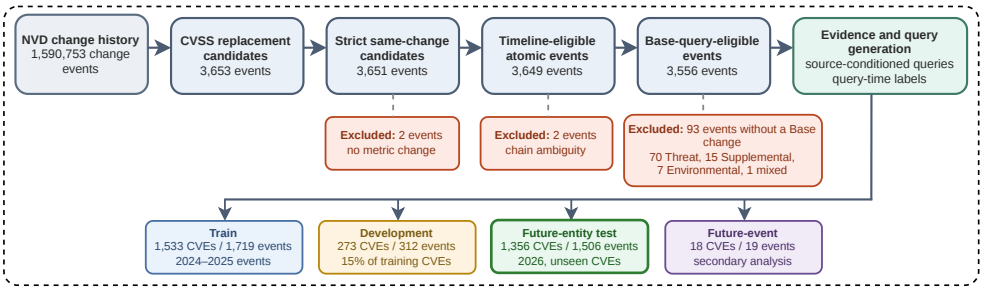


Fig. 3 Dataset construction and evaluation pipeline.

In the change history, 3,653 candidates containing a one-to-one removal and addition for the same CVSS type within the same event were identified. Of these, 3,651 candidates that met the structural requirements were accepted as STRICT candidates. At the candidate level, 2 records were excluded because the old and new vectors were identical and only the evaluator name had changed. During timeline reconstruction, 2 additional events were excluded from the main dataset because of chain ambiguity and were recorded together with the reasons for exclusion. After filtering, 3,649 labelable atomic replacement events and 3,249 vulnerabilities remained. The distribution of candidate events across years (2024: 781, 2025: 1,306, and 2026: 1,564) and CVSS versions (3.1: 3,179, 4.0: 387, 2.0: 68, and 3.0: 17) reflects the acceleration of the

NVD assessment process after 2024 and the transition to CVSS 4.0. Since the primary queries request the current CVSS-B assessment, query-eligible events were restricted to atomic replacement events involving a change in at least one Base metric. In 93 events involving only Threat metrics, Supplemental metrics, Environmental metrics, or multiple non-Base metric groups, the Base vector, CVSS-B score, and severity rating remained unchanged. These events were not removed from the splits but were excluded from query generation and retained in a separate file with the reasons for exclusion because they did not produce a new reference answer. Ultimately, 3,556 events were eligible for query generation.

The active-interval boundaries in Equation (5) are generated from atomic replacement events on the timeline. These event-based labels are programmatically derived from official records, and each label can be traced back to a specific replacement event. The labels have not been manually verified, and their accuracy depends on the accuracy of the NVD records. Therefore, they are referred to as reference labels throughout the paper. Label generation is strictly separated from model inputs; the label-only fields appear solely in the labeling and evaluation files and are listed in Section 5.2.

4.3 Query generation, candidate pools, and quality assurance

The practical application scenario involves developing the method using historical data and applying it to vulnerabilities emerging in the future. By using CVE as the splitting unit, all chains, intervals, events, and queries associated with a specific vulnerability are kept within the same split, with period membership determined by the time of replacement events. While the training and development sets cover events from 2024–2025, the main test set consists exclusively of events from 2026. Furthermore, 18 vulnerabilities that also had events during 2024–2025 were excluded from the test set to ensure that it comprised vulnerabilities unseen during training. This configuration is referred to as a future-entity temporal test based on CVEs unseen during training. Before filtering, the future-entity split consists of 1,392 CVEs and 1,544 atomic events, whereas the primary query evaluation is based on 1,356 CVEs and 1,506 events involving Base metric changes. In the remainder of the paper, the future-entity split refers to the broader event set, while the main future-entity test refers to the subset containing 1,506 queries. A future-event cohort containing 19 future revisions of vulnerabilities observed during the training period is kept separate and reported only in secondary analyses. The development set was selected from 15% of the training-period vulnerabilities using a fixed seed, and the test set was not used for model, feature, threshold, or hyperparameter selection. The primary evidence set is constructed from intervals across all source chains of vulnerabilities with at least one event eligible for query generation. This ensures that the previous and subsequent assessments in the target chain, as well as other chains belonging to the same vulnerability, are included in the same pool. Each evidence record includes the fields from Equation (1), the calculated CVSS-B score, the severity rating, and, if available, the CVSS-BT score as separate metadata. The evidence text is generated using a fixed template based solely on structured fields (source, vulnerability, CVSS version, Base vector, CVSS-B score, severity rating, and observation date). For left-censored records, the text indicates that

the record was active at the beginning of the observation window rather than specifying a date. These texts do not contain labels, subsequent replacement events, threat scores, or scores reported by the source. Source-conditioned queries (Equation (2)) are generated using four template variants from a single family: the current Base metric vector, the current CVSS-B score, the current severity rating, and a combination of these fields. Query time is selected using a deterministic rule as the earlier of thirty days after the replacement event and the midpoint of the remaining active interval. This ensures that both the current and superseded assessments in the target chain are present in the candidate pool. Reference answers are programmatically derived from the evidence active in the target chain at query time. For each of the 3,556 events eligible for query generation, one main post-replacement query was generated, together with 3,555 pre-replacement control queries. Of the 7,111 queries, 3,537 main queries were used in the primary evaluation (training: 1,719, development: 312, and test: 1,506). The 19 queries in the future-event cohort were kept separate, and the control queries were not included in the main results. The base score and severity rating of each evidence interval were calculated from the metric vector using the standard formulas for CVSS 2.0 and 3.x and the official MacroVector method for CVSS 4.0 [14]. For CVSS 4.0, the primary score is the CVSS-B score calculated using only the Base metrics. The calculators were compared with the scores published by the NVD. No discrepancies were found among 198,767 CVSS 2.0 and 3.x assessments. For CVSS 4.0, differences in only 21 of 36,207 assessments resulted from decimal-rounding behavior at binary floating-point boundaries. These differences were resolved, leaving no unexplained records. Temporal integrity is ensured through two automated checks. Pre-replacement control queries are used to validate the labels. Among the 3,536 control queries for which candidate pools were generated, the number of visibility violations was 0, and the number of instances in which replacement evidence observed after query time leaked into the pool was 0. The fields used for label generation are stored separately from the model inputs, and static and dynamic tests verify that the method code does not read these fields (Section 5).

4.4 Dataset statistics

Figure 3 summarizes the construction stages and the temporal split, and two data layers should be distinguished in it. At the level of all eligible atomic events, the training, development, test, and future-event splits contain 1,578 / 1,768, 279 / 318, 1,392 / 1,544, and 18 / 19 CVEs / events, respectively, with left-censored shares of 12.8%, 11.6%, 7.0%, and 0.0%; the 3,630 main events and the 19 future-event records together cover 3,249 unique CVEs. The Base-query-eligible subset shown in the final boxes of the figure excludes the 93 events without a Base change and yields the 1,533 / 1,719, 273 / 312, 1,356 / 1,506, and 18 / 19 CVEs / events that serve as main queries. The main evidence set consists of 9,015 evidence records from all source chains of vulnerabilities with at least one Base-query-eligible event. Retrieval is performed on the full corpus. Of the 265,965 evidence intervals in the timeline, 265,900 were retained, while 65 were excluded, comprising 64 intervals from ambiguous chains and 1 interval with an unparseable vector. The full corpus contains 164,394 unique CVEs and 427 distinct source identifiers, and 1,224 intervals are left-censored.

5 SupersedeRank

In this section, the source-chain and query-time information established in the previous sections is translated into a reranking decision. SupersedeRank is not a new retrieval model or language model. Rather, it is an explainable reranking layer that operates on candidates returned by BM25 and bases each decision solely on fields observable by query time. This layer simultaneously aims to preserve current evidence and to demote evidence from the top ranks if it is inferred to have been superseded by a newer and different assessment visible by query time within the target chain. Prioritizing preservation when these two objectives conflict ensures that current assessments from other sources are not suppressed. In this study, the primary method is designated as SupersedeRank-Rule-Target, while the scope variant is designated as SupersedeRank-Rule-Corpus.

The source-conditioned query (Equation (2)) is applied to the evidence pool visible at query time, BM25 retrieves up to fifty candidates, and SupersedeRank reranks these candidates to produce a refined context consisting of the top-ten candidates. Passing this context to a generation model is the intended use of the system; however, end-to-end RAG evaluation was not performed in this study and remains outside its scope.

5.1 Query-time-visible evidence pool

A single visibility rule applies to all components: a piece of evidence is visible only if its observed activation time is less than or equal to query time. While the retrieval pool is constructed according to Equation (3), the reranker operates on an evidence store constrained by the same rule. An assessment observed after query time cannot serve as a candidate, nor can it serve as evidence that another candidate has been superseded. For left-censored records, the observed activation time is treated merely as a lower bound, and the method does not derive age from this timestamp. The evidence store indexes candidates based on the chain key defined in Equation (4), while the query's target chain is determined directly by the triplet of CVE ID, source, and CVSS version, without requiring label information.

5.2 SupersedeRank-Rule-Target

SupersedeRank-Rule-Target reduces two conditions to a single inferred-supersession signal, considering a candidate superseded only if a newer and different Base assessment within the target chain is visible at query time. Figure 4 summarizes this decision process for a BM25 candidate. A candidate is preserved if it does not belong to the query's target chain or if no newer witness assessment in the same chain is visible at query time. If a witness is found, the canonical vectors of the two assessments, comprising only the Base metrics, are compared. If the vectors are identical, the candidate is preserved because re-adding the same value does not constitute a replacement. If they differ, the candidate is inferred to be superseded and moved to the inferred-superseded block. In the final ranking, the protected block appears first and the inferred-superseded block appears after it, while the original BM25 order is maintained within each block. Only evidence records from the main dataset serve as

witnesses in this process. Cross-event candidates obtained by matching removal and addition records across different events are not accepted as witnesses.

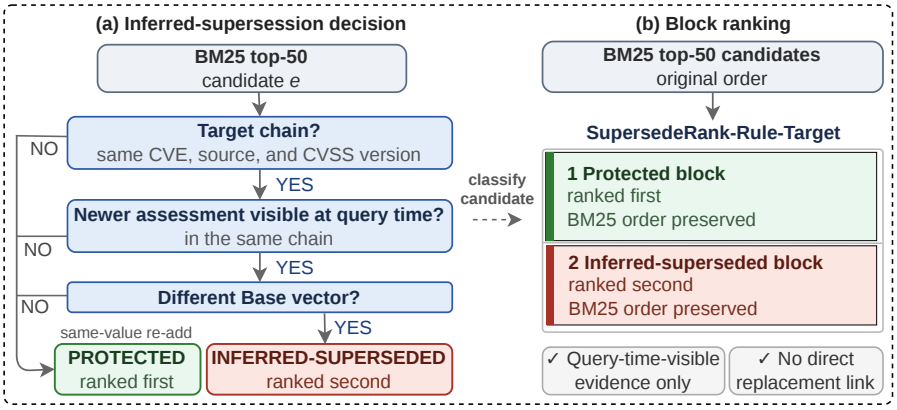


Fig. 4 SupersedeRank-Rule-Target decision workflow.

This decision is a special case of the function f in Equation (7): $p(e, q)$ is a binary signal, and f uses this signal to determine the block order while using $r(e, q)$ to preserve the within-block order. The decision logic reads only fields observable by query time, including the CVE ID, source ID, CVSS version, observed activation time, query time, BM25 rank, Base vectors of visible candidates, the presence of newer visible evidence in the same chain, differences between the visible Base vectors, and the left-censoring flag. The end of the active interval, supersession links, replacement type, timeline status, the ID of the event that generated the label, CURRENT or OUTDATED labels, events occurring after query time, and derived fields directly encoding this information are not used as inputs. The method contains no learned parameters, thresholds, or training and is deterministic and reproducible. It does not require an explicit supersession link but requires access to the source chain, observed activation time, Base vector fields, and query-time-visible records within the same chain.

5.3 SupersedeRank-Rule-Corpus

The same signal can be applied to all fifty candidates without being restricted to the target chain. This variant, termed SupersedeRank-Rule-Corpus, performs temporal cleanup across the entire candidate pool by moving a candidate to the second block if it is inferred to have been superseded by a newer and different assessment visible at query time within its own chain, even if the candidate belongs to a different vulnerability. The two conditions differ in their suppression scope. In the SupersedeRank-Rule-Target condition, the rule is applied only to the chain targeted by the query. This chain is determined from the query's source field and does not depend on reference labels. In the SupersedeRank-Rule-Corpus condition, target-chain information is not used, and each candidate is evaluated within its own chain. This condition is closer to queries

that do not specify a target chain, but it may leave fewer than ten candidates in the first block for some queries. Section 6.4 evaluates the trade-off between these two suppression scopes.

5.4 Complexity, leakage prevention, and evaluation safeguards

The computational cost per query consists of a dictionary lookup using the chain key for fifty candidates and a linear scan over query-time-visible records in the target chain. The chain index is constructed only once, and the method, unlike model-based rerankers, imposes negligible overhead on the retrieval process. Measured execution times are presented in Section 6.1. Static and dynamic checks verified that the reranking decision is independent of label information. Static checks confirmed through source code analysis that the rule module does not import the label package and that forbidden field names are not used in the decision logic. Dynamic checks confirmed, using both synthetic examples and the actual evidence store, that the output remains unchanged when forbidden fields are randomly corrupted, labels are shuffled, or artificial witnesses observed after query time are added. A policy using the direct replacement link was also implemented. However, because this policy relies on information used to generate the label, it is reported as a diagnostic comparison called the Direct-Link Policy rather than as the primary method. Consequently, this policy is not considered an upper bound for the primary method. The evaluation safeguards consist of five fundamental rules: the query-time cutoff rule, the source-conditioned chain, the shared BM25 candidate pool, the exclusion of reference labels from the input, and the exclusion of direct replacement links from the input. Reference labels are used solely for evaluation and are not included in the reranker's input. The extent to which the method achieves its two objectives under these safeguards is evaluated in the next section.

6 Experiments and Results

All analyses are conducted at the CVE level because queries associated with the same vulnerability are not independent.

6.1 Experimental setup, comparison methods, and metrics

The primary statistical evaluation is based only on the main future-entity test, which consists of 1,506 post-replacement queries for 1,356 vulnerabilities that were never observed during the training period. The training set (1,719 queries) and development set (312 queries) are used only for development and descriptive reporting, while the future-event cohort (19 queries) is excluded from all statistical analyses. Retrieval is performed over the full corpus. For each query, BM25 ($k_1 = 1.5$, $b = 0.75$) is applied to the evidence intervals visible at query time among the 265,900 accepted intervals, and the top fifty candidates are selected. Candidates with equal scores are ordered using a fixed hash value that does not incorporate temporal information. The same set of fifty candidates is shared across all rerankers. BM25 retrieves the target CURRENT evidence among the top fifty candidates for all main queries and the target OUTDATED

evidence for 99.97% of the queries. Because there are no retrieval failures for the target CURRENT evidence in the shared BM25 candidate pool, Recall@50 is 1.0 for all methods. No method was tuned using the test set. All steps involving randomness are executed using fixed seeds. Because queries associated with the same vulnerability are not independent, CVE-level cluster bootstrap is applied. The 1,356 vulnerabilities in the test set are sampled with replacement, and all queries associated with each selected vulnerability are included together. A total of 10,000 iterations are used, and 95% confidence intervals are calculated using the percentile method. The same bootstrap samples are used across all methods and paired comparisons. Two-sided bootstrap p-values are calculated for the paired differences between SupersedeRank-Rule-Target and each comparison method. Holm correction is applied within the prespecified primary family, which consists of four comparisons and three metrics: Current-At1-And-No-Stale-At10, Stale-Evidence-Rate@10, and graded NDCG@10. The results are interpreted together with effect sizes and confidence intervals. All experiments were conducted on a workstation equipped with a 24-core processor, 31.7 GB of memory, and an NVIDIA GeForce RTX 3070 Ti GPU with 8 GB of memory. SupersedeRank-Rule-Target built the chain index for 265,965 evidence records in four seconds and reranked all main queries in under two seconds, adding negligible latency to the retrieval pipeline.

Three methods are reported at the retrieval level. BM25 generates the primary candidate pool. Dense retrieval, based on the cosine similarity of bge-base-en-v1.5 embeddings, retrieves the current evidence for the target chain within the top fifty candidates for 67.5% of the main queries. Hybrid retrieval combines the BM25 and dense retrieval lists through reciprocal rank fusion ($k = 60$) and achieves this for 99.5% of the queries. Since BM25 retrieves the current evidence for all queries, the shared candidate pool is constructed using BM25, while the dense and hybrid results are retained as descriptive comparisons. At the reranking level, BM25 preserves the initial ranking. BM25+Recency adds a recency bonus that decays exponentially with observed age to the normalized BM25 score. Its parameters were selected only on the development set ($\tau = 30$ days, $\lambda = 0.5$). The Cross-encoder is a semantic reranker without temporal information and uses the raw logit score of an MS MARCO-trained MiniLM model. The primary method is SupersedeRank-Rule-Target, and SupersedeRank-Rule-Corpus is its scope variant. The Direct-Link Policy, which uses explicit replacement links, is presented separately as a diagnostic policy. The Temporal-clean oracle, constructed using the reference temporal labels, is presented as the true upper bound. These methods are reported under the Diagnostic policy and Oracle upper bound groups, respectively. Ranking quality is evaluated using MRR@10 and graded NDCG@10, for which the target CURRENT evidence has a relevance grade of 2, OTHER_SOURCE_CURRENT evidence a grade of 1, and all other evidence a grade of 0. Stale-Evidence-Rate@10 measures the proportion of target-chain OUTDATED evidence among the top-ten candidates. Outdated-Suppression@10, Current-Preservation@10, and the composite metric Current-At1-And-No-Stale-At10, which requires the current evidence to be ranked first with no superseded target evidence in the top ten, measure the primary contribution.

6.2 Main ranking results

Table 1 presents the point estimates and 95% confidence intervals for the main future-entity test. The following paragraphs discuss these findings for RQ1 and RQ2; RQ3 and RQ4 are addressed in Sections 6.4 and 6.3.

Table 1 Future-entity test results.

Method	Cur@1∧ NoStale@10	Stale @10	gNDCG @10	MRR @10	OutSupp @10	CurPres @10
Retrieval and semantic baselines						
BM25	0.000 0.000–0.000	0.105 0.104–0.106	0.793 0.785–0.801	0.721 0.709–0.734	0.000 0.000–0.000	1.000 1.000–1.000
BM25+Recency	0.000 0.000–0.000	0.105 0.104–0.106	0.961 0.958–0.964	0.988 0.984–0.991	0.000 0.000–0.000	1.000 1.000–1.000
Cross-encoder	0.079 0.065–0.093	0.063 0.060–0.066	0.520 0.500–0.539	0.505 0.483–0.527	0.400 0.374–0.425	0.775 0.753–0.796
SupersedeRank						
SupersedeRank-Rule-Target	0.998 0.995–1.000	0.000 0.000–0.000	0.996 0.995–0.997	0.999 0.998–1.000	1.000 1.000–1.000	1.000 1.000–1.000
SupersedeRank-Rule-Corpus	0.834 0.815–0.853	0.017 0.015–0.019	0.999 0.999–1.000	0.999 0.998–1.000	0.835 0.816–0.854	1.000 1.000–1.000
Diagnostic policy						
Direct-Link Policy (diagnostic)	0.831 0.812–0.851	0.017 0.015–0.019	0.998 0.997–0.999	0.998 0.996–0.999	0.835 0.816–0.854	1.000 1.000–1.000
Oracle upper bound						
Temporal-clean oracle (oracle upper bound)	1.000 1.000–1.000	0.000 0.000–0.000	1.000 1.000–1.000	1.000 1.000–1.000	1.000 1.000–1.000	1.000 1.000–1.000

The answer to RQ1 is negative. Although BM25 retrieves the target CURRENT evidence among the top fifty candidates for all main queries, it assigns the same score to the current and superseded evidence in 87.3% of the queries and ranks the superseded evidence first in 56.6% of them. The cross-encoder, despite its strong semantic capabilities, fails to increase the rate at which current evidence appears in the top position and pushes current evidence outside the top ten in approximately one-fifth of the queries. Although BM25+Recency ranks the current evidence first in 97.5% of the test queries, it does not improve Stale-Evidence-Rate@10 over BM25 and merely moves the superseded evidence to the second position. The answer to RQ2 is positive. In the main future-entity test, SupersedeRank-Rule-Target reached 0.998 on the composite metric with complete suppression and preservation (Table 1). Across all main queries, superseded evidence remained in the top ten in only two cases. In both cases, the required witness assessment belonged to the cross-event records excluded from the primary scope, reflecting a deliberate design decision. Current evidence was not suppressed in any query, and current assessments from other sources were not penalized.

In the future-entity test, only SupersedeRank-Rule-Target combines complete current-evidence preservation with complete suppression of superseded target evidence from the top-ten context.

Paired test differences between SupersedeRank-Rule-Target and each comparison method were computed for the three primary-family metrics as mean differences with confidence intervals, two-sided p-values, Holm-adjusted p-values, and query-level counts of wins, losses, and ties; the complete table, including the secondary and unadjusted metrics, is released together with the dataset (see Code and data availability). For Stale-Evidence-Rate@10, a negative difference favors SupersedeRank-Rule-Target. All twelve tests in the primary family were significant after Holm correction ($p_{\text{Holm}} = 0.0012$; bootstrap resolution of 10^{-4}). The difference in composite metric relative to both BM25 and BM25+Recency was 0.998 (0.995–1.000), while the difference in Stale-Evidence-Rate@10 was −0.105. The rate of superseded evidence decreased in all 1,506 test queries. Compared with the cross-encoder, the difference in the composite metric was 0.919, with SupersedeRank-Rule-Target winning on 1,384 queries and losing on none. Relative to SupersedeRank-Rule-Corpus, the difference in the composite metric was 0.164 (0.145–0.183) and the difference in Stale-Evidence-Rate@10 was −0.017, whereas graded NDCG@10 was lower by 0.004 (−0.005 to −0.003; 0 wins, 106 losses, 1,400 ties), which is the only primary comparison in which the scope variant is ahead. Although the difference in MRR@10 relative to BM25+Recency was small (0.011), this result shows that the primary contribution lies not in top-rank accuracy but in preserving current evidence while removing superseded evidence from the top-ten context.

6.3 Construct validity and query-time sensitivity

The top-rank performance of BM25+Recency must be evaluated in the context of a structural consequence of the query generation design. For the main queries, query time is at most thirty days after the replacement event, and this distance is exactly thirty days for 90% of the queries. In every post-replacement query, the newest visible evidence in the target chain is structurally CURRENT. Since BM25 assigns the same score to current and superseded evidence in 87.3% of the queries, any recency bonus that decays with age resolves these ties in favor of the current evidence. The bonus, however, changes only the ranking and does not move the superseded evidence outside the top-ten candidates. To measure this dependency, the query-time distance after the replacement event was varied using fixed values, and the results are presented in Table 2. The table reports, for each fixed distance, the number of queries that remain valid, the rate at which BM25 and BM25+Recency rank the current evidence first, and the MRR@10 and Stale-Evidence-Rate@10 of BM25+Recency. Queries were excluded if the active interval ended before the new distance or if the resulting query time fell outside the observation window. The BM25+Recency parameters were kept unchanged. When the distance was increased from one day to 365 days, the proportion of queries for which BM25+Recency ranked the current evidence first decreased from 0.989 to 0.810, whereas Stale-Evidence-Rate@10 remained unchanged at all distances. SupersedeRank-Rule-Target does not use a recency bonus based on the absolute age of evidence. Instead, it bases its decision on a newer and different assessment in the

same chain that is visible at query time. The distance analysis covers only BM25 and the recency-based methods; SupersedeRank-Rule-Target was not re-evaluated at these distances. The query-time-visible corpus also grows over time, from an average of 88,821 evidence records for training queries to 206,254 records for test queries. However, the shared BM25 candidate pool contains the target CURRENT evidence in both periods. The top-rank performance of the recency-based comparison method should therefore be interpreted as an optimistic estimate specific to the query-time rule used in this dataset. The first part of RQ4 was addressed using the test results above: within the future-event cohort of 19 queries, SupersedeRank-Rule-Target reached 1.000 on the composite metric, although the cohort is too small for this result to be more than descriptive. The negative controls behaved as designed. In the 3,536 pre-replacement control queries, the target evidence had not yet been superseded, the rule generated no signal, and the ranking remained identical to that of BM25; likewise, no rule signal was generated for the 7,672 records in which the same value was removed and re-added, because there was no vector difference.

Table 2 Query-time offset sensitivity of the recency bonus (main queries).

Offset	n	BM25 Top-1	BM25+Recency Top-1	BM25+Recency MRR@10	BM25+Recency Stale@10
+1d	3451	0.424	0.989	0.994	0.105
+30d	3286	0.424	0.957	0.979	0.104
+90d	3042	0.417	0.884	0.942	0.104
+180d	2362	0.411	0.875	0.937	0.105
+365d	1402	0.400	0.810	0.904	0.105

6.4 Rule scope and error analysis

For RQ3, Table 3 compares the SupersedeRank-Rule-Target and SupersedeRank-Rule-Corpus variants using the 3,537 main queries. SupersedeRank-Rule-Corpus moves 50,260 candidate instances to the second block. Of these, only 3,709 belong to the query's target chain, while the remainder are records inferred to be superseded within the chains of other vulnerabilities. SupersedeRank-Rule-Target moves only these 3,709 candidate instances. The broader cleanup provides a slight gain in graded NDCG@10 (0.999 vs. 0.994) because current assessments from other sources are ranked ahead of superseded irrelevant records. In contrast, for 284 queries, fewer than ten candidates remain in the protected block, forcing superseded target evidence back into the top ten. The composite metric value of 0.834 for SupersedeRank-Rule-Corpus in the main future-entity test results from this limitation. When the test queries are split by capacity, the two conditions produce equivalent results (0.998 vs. 0.998) for the 1,258 queries with at least ten protected candidates. For the 248 queries with fewer than ten protected candidates, SupersedeRank-Rule-Corpus scores 0.000, whereas SupersedeRank-Rule-Target scores 1.000. For both methods, the loss of current evidence, evidence from other sources, and graded relevance is zero. At the

query level, the two variants produce identical rankings for 504 of the main queries; SupersedeRank-Rule-Target is better on the composite metric for 284 queries and SupersedeRank-Rule-Corpus for 7, while the remaining 2,742 queries are tied on the composite and stale metrics. Since the primary method was selected according to criteria established before observing the results, including direct focus on the target chain, minimal interference with semantic ranking, absence of capacity constraints, and simpler explanation, SupersedeRank-Rule-Corpus is reported only as an analysis of suppression scope.

Table 3 Comparison of suppression scopes (main queries).

	SupersedeRank-Rule-Corpus	SupersedeRank-Rule-Target
Suppressed candidates (total)	50260	3709
Suppressed outside the target chain	46551	0
Queries with < 10 preserved candidates	284	0
Cur@1$\wedge$NoStale@10	0.917	0.995
Stale@10	0.008	0.000
OutSupp@10	0.919	0.999
gNDCG@10	0.999	0.994
MRR@10	0.999	0.998

The error analysis distinguishes among three sources of error. Since there are no retrieval errors for the target CURRENT evidence in the shared BM25 top-50 candidate pool, losses within this pool arise entirely from reranking decisions. Retrieval losses observed in dense retrieval are outside the main comparison. Cases in which the rule cannot make a decision correspond to queries without a visible witness in the target chain. However, because the new assessment is visible in every post-replacement query, no such cases remain for SupersedeRank-Rule-Target. Queries involving multiple source chains pose no additional challenge because evidence from other chains is marked only as `OTHER_SOURCE_CURRENT` and is not suppressed. Left-censored target evidence is not present in the main queries, so left-censoring does not affect the decisions.

7 Discussion and Limitations

When the experimental findings are evaluated collectively, they clearly demonstrate that the core problem lies not in the retrieval stage but in the validity decision. This distinction explains why components measuring semantic relevance fail to detect validity and why a single rule based on the source chain suffices.

7.1 Main findings and practical implications

Semantic retrieval reliably retrieves the target evidence. The fundamental issue is not locating the target vulnerability but distinguishing between current and superseded

assessments within the same source chain. Because dense retrieval fails to match identifier fields through lexical matching, it misses the target evidence in approximately one-third of the queries. Although hybrid merging largely compensates for this loss, it does not match the coverage of BM25. The cross-encoder, unable to resolve temporal validity, may lose current evidence while only partially suppressing superseded evidence, demonstrating that semantic strength cannot substitute for validity information. Although the combination of BM25 and a recency bonus succeeds in promoting current evidence to the top rank, this capability is directly linked to the query generation structure, is sensitive to the query-time distance, and fails to remove superseded evidence from the top-ten context. SupersedeRank-Rule-Target achieves two objectives simultaneously. It preserves current evidence while moving superseded evidence within the target chain outside the top-ten candidates. Although SupersedeRank-Rule-Corpus performs more extensive cleanup and slightly improves graded relevance, it leaves fewer than ten candidates in the first block for certain queries. Consequently, the scope of suppression is a design choice that affects the two objectives in opposing ways. Since Direct-Link Policy, which uses explicit replacement links, is also subject to a similar capacity effect, it cannot be considered an upper bound for the primary method. Instead, the Temporal-clean oracle represents the true upper bound.

Since the experiments measure retrieval and reranking quality, the implications are limited to this level. SupersedeRank-Rule-Target is an applicable reranking policy for evolving information systems with an explicit source-chain structure and query-time visibility. However, the current experimental findings are limited to atomic CVSS Base revisions. In a vulnerability knowledge base containing source-chain and observed activation time fields, SupersedeRank-Rule-Target can be integrated into the retrieval pipeline without an additional model or training. It improves the temporal composition of the evidence context by moving superseded assessments outside the top-ten candidates while preserving current assessments from the target and other sources. Each decision can be explained by presenting the witness evidence and the vector difference. Whether this context improves answer accuracy, reduces hallucinations, or improves security decisions has not been measured and is not claimed.

7.2 Limitations

The limitations concern the labels, the data partitioning, and the scope of the evaluation. The labels have not been manually verified. They are event-based labels derived programmatically from explicit old-to-new value transitions, and their accuracy depends on the accuracy of the NVD change history records. The primary labels rely solely on high-confidence atomic CVSS replacement events. Changes in vulnerability descriptions, affected product configurations, mitigation recommendations, patch status, and affected version information are outside the scope. The actual start time of left-censored assessments is unknown, and the lower bound of the observation window was not used as a proxy for this time. The normalization of source aliases and chains may introduce uncertainty. Therefore, two events were excluded from the main dataset, and 131 cross-event candidates were not included in the primary results. The future-event cohort is small, comprising only 18 vulnerabilities and 19 events. Since the source distribution in the test set differs from that in the training set, the effects

of generalization to unseen vulnerabilities and source shift cannot be fully disentangled. Query time for the main queries is at most thirty days after the replacement event. This design provides a structural advantage to the recency-based comparison method (Section 6.3), and the main results are reported for this single query-time rule. SupersedeRank-Rule-Target requires source-conditioned chain information. Therefore, only SupersedeRank-Rule-Corpus can be applied to queries without a source field. However, this variant is affected by capacity limitations in 284 main queries. The distance analysis was performed only for BM25 and recency-based methods. The comparison does not include a learned or hybrid reranker; the study is restricted to the parameter-free rule and its scope variant. The study measures retrieval and reranking quality but does not evaluate the effect of the cleaned context on answer quality or cross-domain generalization, such as to HoH. For CVSS 4.0, the score published by the NVD corresponds to the full-vector score when the source reports a Threat metric. However, the Base score in the evidence records was calculated using only the Base metrics, and this difference was documented. The findings within these limitations support the general conclusion summarized in the final section.

8 Conclusion

This study addressed the problem of superseded evidence in RAG systems within a traceable setting in vulnerability management. The validity of a CVSS assessment is not a static attribute of a record but depends on the relationship between the record and query time. In the dataset created for this study, this relationship was derived from atomic replacement events in the NVD change history without additional human annotation and evaluated on vulnerabilities unseen during training using a future-entity test protocol. Through a single decision based solely on fields observable by query time, SupersedeRank-Rule-Target preserved current evidence while almost entirely removing superseded target evidence from the top-ten context. In contrast, the recency bonus and semantic reranking could not achieve both objectives simultaneously. Future work could investigate free-text description revisions, corpus-level scope variants without capacity constraints, evaluations on out-of-domain datasets, and the effect of the cleaned context on downstream answer quality.

Statements and Declarations

- **Funding:** No funding was received for conducting this study.
- **Competing interests:** The author has no relevant financial or non-financial interests to disclose.
- **Ethics approval:** Not applicable.
- **Consent to participate:** Not applicable.
- **Consent for publication:** Not applicable.
- **Code and data availability:** The raw data were obtained from the public CVE and change-history interfaces of the NVD. The source code, the derived dataset, and the per-query results of all reported runs are available at https://github.com/Doygun/SupersedeRank.

- **Author contribution:** Doygun Demirol performed the conceptualization, methodology, software development, data curation, formal analysis, investigation, validation, visualization, and preparation and revision of the manuscript.

References

[1] Anwar A, Abusnaina A, Chen S, et al (2022) Cleaning the NVD: Comprehensive quality assessment, improvements, and enhancements. IEEE Transactions on Dependable and Secure Computing 19(6):4255–4269. https://doi.org/10.1109/TDSC.2021.3125270

[2] Campos R, Dias G, Jorge AM, et al (2014) Survey of temporal information retrieval and related applications. ACM Computing Surveys 47(2):15:1–15:41. https://doi.org/10.1145/2619088

[3] Chen HT, Zhang M, Choi E (2022) Rich knowledge sources bring complex knowledge conflicts: Recalibrating models to reflect conflicting evidence. In: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, pp 2292–2307, https://doi.org/10.18653/v1/2022.emnlp-main.146

[4] Chen Z, Min E, Zhao X, et al (2025) A question answering dataset for temporal-sensitive retrieval-augmented generation. Scientific Data 12(1):1855. https://doi.org/10.1038/s41597-025-06098-y

[5] Chopra S, Ahmad H, Goel D, et al (2026) ChatNVD: Advancing cybersecurity vulnerability assessment with large language models. IEEE Access 14:18595–18610. https://doi.org/10.1109/ACCESS.2026.3659205

[6] Cormack GV, Clarke CLA, Buettcher S (2009) Reciprocal rank fusion outperforms Condorcet and individual rank learning methods. In: Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, pp 758–759, https://doi.org/10.1145/1571941.1572114

[7] Croft R, Babar MA, Kholoosi MM (2023) Data quality for software vulnerability datasets. In: Proceedings of the 45th IEEE/ACM International Conference on Software Engineering (ICSE). IEEE, pp 121–133, https://doi.org/10.1109/ICSE48619.2023.00022

[8] Dhingra B, Cole JR, Eisenschlos JM, et al (2022) Time-aware language models as temporal knowledge bases. Transactions of the Association for Computational Linguistics 10:257–273. https://doi.org/10.1162/tacl_a_00459

[9] Dong A, Zhang R, Kolari P, et al (2010) Time is of the essence: Improving recency ranking using twitter data. In: Proceedings of the 19th International Conference on World Wide Web (WWW). ACM, pp 331–340, https://doi.org/10.1145/

1772690.1772725

[10] Dong Y, Guo W, Chen Y, et al (2019) Towards the detection of inconsistencies in public security vulnerability reports. In: Proceedings of the 28th USENIX Security Symposium. USENIX Association, pp 869–885

[11] Elbaz C, Rilling L, Morin C (2020) Fighting n-day vulnerabilities with automated CVSS vector prediction at disclosure. In: Proceedings of the 15th International Conference on Availability, Reliability and Security (ARES). ACM, pp 1–10, https://doi.org/10.1145/3407023.3407038

[12] Fayyazi R, Hoyos Trueba S, Zuzak M, et al (2025) ProveRAG: Provenance-driven vulnerability analysis with automated retrieval-augmented LLMs. IEEE Access 13:212815–212826. https://doi.org/10.1109/ACCESS.2025.3638251

[13] FIRST.Org, Inc. (2019) Common vulnerability scoring system version 3.1: Specification document (revision 1). Tech. rep., Forum of Incident Response and Security Teams, https://www.first.org/cvss/v3.1/specification-document

[14] FIRST.Org, Inc. (2023) Common vulnerability scoring system version 4.0: Specification document. Tech. rep., Forum of Incident Response and Security Teams, https://www.first.org/cvss/v4.0/specification-document

[15] Jacobs J, Romanosky S, Edwards B, et al (2021) Exploit prediction scoring system (EPSS). Digital Threats: Research and Practice 2(3):20:1–20:17. https://doi.org/10.1145/3436242

[16] Jang J, Ye S, Lee C, et al (2022) TemporalWiki: A lifelong benchmark for training and evaluating ever-evolving language models. In: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, pp 6237–6250, https://doi.org/10.18653/v1/2022.emnlp-main.418

[17] Johnson P, Lagerström R, Ekstedt M, et al (2018) Can the common vulnerability scoring system be trusted? A Bayesian analysis. IEEE Transactions on Dependable and Secure Computing 15(6):1002–1015. https://doi.org/10.1109/TDSC.2016.2644614

[18] Kanhabua N, Blanco R, Nørvåg K (2015) Temporal information retrieval. Foundations and Trends in Information Retrieval 9(2):91–208. https://doi.org/10.1561/1500000043

[19] Kasai J, Sakaguchi K, Le Bras R, et al (2023) RealTime QA: What's the answer right now? In: Advances in Neural Information Processing Systems, pp 49025–49043, https://doi.org/10.52202/075280-2130

[20] Kortukov E, Rubinstein A, Nguyen E, et al (2024) Studying large language model behaviors under context-memory conflicts with real documents. In: Proceedings of the First Conference on Language Modeling (COLM), openReview xm8zYRfrqE

[21] Kühn P, Relke DN, Reuter C (2023) Common vulnerability scoring system prediction based on open source intelligence information sources. Computers & Security 131:103286. https://doi.org/10.1016/j.cose.2023.103286

[22] Lewis P, Perez E, Piktus A, et al (2020) Retrieval-augmented generation for knowledge-intensive NLP tasks. In: Advances in Neural Information Processing Systems, pp 9459–9474

[23] Li F, Paxson V (2017) A large-scale empirical study of security patches. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS). ACM, pp 2201–2215, https://doi.org/10.1145/3133956.3134072

[24] Li X, Croft WB (2003) Time-based language models. In: Proceedings of the Twelfth International Conference on Information and Knowledge Management (CIKM). ACM, pp 469–475, https://doi.org/10.1145/956863.956951

[25] Liu TY (2009) Learning to rank for information retrieval. Foundations and Trends in Information Retrieval 3(3):225–331. https://doi.org/10.1561/1500000016

[26] Longpre S, Perisetla K, Chen A, et al (2021) Entity-based knowledge conflicts in question answering. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, pp 7052–7063, https://doi.org/10.18653/v1/2021.emnlp-main.565

[27] Meem JA, Rashid MS, Dong Y, et al (2024) PAT-questions: A self-updating benchmark for present-anchored temporal question-answering. In: Findings of the Association for Computational Linguistics: ACL 2024. Association for Computational Linguistics, pp 13129–13148, https://doi.org/10.18653/v1/2024.findings-acl.777

[28] Ouyang J, Pan T, Cheng M, et al (2025) HoH: A dynamic benchmark for evaluating the impact of outdated information on retrieval-augmented generation. In: Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, Vienna, Austria, pp 6036–6063, https://doi.org/10.18653/v1/2025.acl-long.301

[29] Rashkin H, Nikolaev V, Lamm M, et al (2023) Measuring attribution in natural language generation models. Computational Linguistics 49(4):777–840. https://doi.org/10.1162/coli_a_00486

[30] Reimers N, Gurevych I (2019) Sentence-BERT: Sentence embeddings using siamese BERT-networks. 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). Association for Computational Linguistics, pp 3982–3992, https://doi.org/10.18653/v1/D19-1410

[31] Robertson SE, Zaragoza H (2009) The probabilistic relevance framework: BM25 and beyond. Foundations and Trends in Information Retrieval 3(4):333–389. https://doi.org/10.1561/1500000019

[32] Vu T, Iyyer M, Wang X, et al (2024) FreshLLMs: Refreshing large language models with search engine augmentation. In: Findings of the Association for Computational Linguistics: ACL 2024. Association for Computational Linguistics, pp 13697–13720, https://doi.org/10.18653/v1/2024.findings-acl.813

[33] Wallat J, Jatowt A, Anand A (2024) Temporal blind spots in large language models. In: Proceedings of the 17th ACM International Conference on Web Search and Data Mining (WSDM). ACM, pp 683–692, https://doi.org/10.1145/3616855.3635818

[34] Xu R, Qi Z, Guo Z, et al (2024) Knowledge conflicts for LLMs: A survey. In: Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Miami, Florida, USA, pp 8541–8565, https://doi.org/10.18653/v1/2024.emnlp-main.486

[35] Zerhoudi S, Granitzer M, Mitrović J (2026) NuggetIndex: Governed atomic retrieval for maintainable RAG. In: Proceedings of the 49th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, Melbourne, VIC, Australia, pp 2286–2296, https://doi.org/10.1145/3805712.3809687

[36] Zhang M, Choi E (2021) SituatedQA: Incorporating extra-linguistic contexts into QA. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, pp 7371–7387, https://doi.org/10.18653/v1/2021.emnlp-main.586

[37] Zhou Y, Li W, Lu Y, et al (2026) Mitigating context-memory conflicts in LLMs through dynamic cognitive reconciliation decoding. IEEE Transactions on Audio, Speech and Language Processing 34:1004–1015. https://doi.org/10.1109/TASLPRO.2026.3657980