Check for updates

Journal of KIIT. Vol. 24, No. 10, pp. 325-338, Oct. 31, 2026. pISSN 1598-8619, eISSN 2093-7571
http://dx.doi.org/10.14801/jkiit.2026.24.10.325

자율 모의해킹 에이전트를 위한 다중 검색 융합 컨텍스트 구성 아키텍처 및 전달-행동 격차 분석

양주성*, 박상우**, 김광현***

A Multi-Signal Retrieval Fusion Architecture for Context Construction in Autonomous Penetration-Testing Agents and Its Delivery-Action Gap Analysis

Jusung Yang*, Sang Woo Park**, and Gwang-Hyun Kim***

본 연구는 사사번호 (RQT-26-041198) 이 성과는 정부(과학기술정보통신부)의 재원으로 2026년 고성능컴퓨팅 지원사업의 사용자 지원을 받아 수행된 연구임 This research was supported by the Ministry of Science and ICT (MSIT) through the 2026 High-Performance Computing (HPC) Support Program

요 약

자율 모의해킹 에이전트는 매 턴 필요한 보안 지식을 프롬프트에 배치해야 한다. 그러나 보안 도메인의 어휘 불일치로 단일 검색기는 질의 유형별 성능 편차를 보이며, 기존 평가는 문서 순위나 최종 성공 여부만 측정해 전달 손실과 실행 손실을 구별하지 못한다. 본 연구는 학습 기반 희소 검색, 밀집 임베딩, 백링크 그래프를 가중 상호 순위 융합(RRF)으로 결합한 지식 검색 계층과, 전달 정확도 및 행동 충실도를 분리 측정하는 평가 체계를 제안한다. 검색 경로 조합을 비교한 실험에서 다중 신호 융합은 신호별 약점을 보완하고 가장 취약한 유형의 성능 저하를 완화하는 경향을 보였다. 필수 지식이 모두 전달된 뒤에도 42%의 전달-행동 격차가 관측되어, 지식 검색 외에 기반 언어 모델의 인지 병목이 존재할 가능성을 시사하였다.

Abstract

Autonomous penetration testing agents require relevant security knowledge at each turn. Vocabulary mismatch causes retrieval performance to vary across query types, while existing evaluations fail to distinguish delivery from execution failures. We propose a retrieval layer combining learned sparse retrieval, dense embeddings, and a backlink graph via weighted Reciprocal Rank Fusion (RRF), alongside separate measures of delivery accuracy and action fidelity. Experiments suggest that multi-signal fusion offsets individual weaknesses and improves robustness across query types. A 42% delivery‑action gap persisted despite complete knowledge delivery, suggesting a potential cognitive bottleneck in the underlying language model.

Keywords

penetration-testing agents, context construction, retrieval fusion, delivery-action gap

* 광주대학교 광정보통신학과 박사과정
- ORCID: https://orcid.org/0009-0005-4899-7932
** 광주대학교 음악학과 학사
- ORCID: https://orcid.org/0009-0006-4754-0647
*** 광주대학교 AI소프트웨어학과 교수(교신저자)
- ORCID: https://orcid.org/0009-0003-2824-8331

· Received: Sep. 10, 2026, Revised: Oct. 16, 2026, Accepted: Oct. 19, 2026
· Corresponding Author: Gwang-Hyun Kim
Dept. AI Software, Gwangju University, 277 Hyodbok-ro, Nam-gu, Gwangju, Korea
Tel.: +82-62-670-2459, Email: ghkim@gwangju.ac.kr

Ⅰ. 서 론

자율 모의해킹 에이전트는 탐색, 취약점 확인, 공격 실행을 여러 턴에 걸쳐 수행하며, 도구 실행에서 새로운 단서를 얻을 때마다 다음 턴에서 살펴볼 공격 표면과 사용할 도구가 달라진다. 이에 따라 에이전트는 매 턴 새로 확인한 정보를 바탕으로 다음 도구 호출을 결정한다[1]. 에이전트는 다음 행동을 선택할 때 현재 단서와 관련된 보안 지식을 참고할 수 있지만, 프롬프트의 분량이 제한되어 있어 모든 지식을 한 번에 제공하기는 어렵다. RAG는 외부 지식에서 질의와 관련된 내용을 검색해 프롬프트에 제공하는 방식을 제시하였다[2]. 다만 보안 분야에서는 같은 공격 기법도 '인증 우회'와 'authentication bypass'처럼 다르게 표현되므로, 정확한 단어 일치에 의존하면 관련 지식을 놓칠 수 있다[3]. 본 연구는 매 턴 새로 생성되는 상황 질의를 이용해 필요한 보안 지식을 검색하고, 정해진 프롬프트 예산 안에 배치하는 동적 컨텍스트 구성 방법을 제안한다. 검색 단계에서는 희소 검색, 밀집 검색, 백링크 그래프를 결합해 표현이 다른 질의와 정확한 식별자 질의를 함께 처리한다. 평가 단계에서는 정답 지식이 프롬프트에 포함되었는지와 그 지식이 올바른 도구 호출로 이어졌는지를 분리해 측정한다.

1.1 기존 연구와의 차별점

본 연구와 기존 연구의 차이는 평가 대상에 있다. 기존 검색 연구는 문서 순위를, RAG 평가는 생성 답변의 품질을, 모의해킹 에이전트 연구는 최종 과제 성공을 주로 측정했다면, 본 연구는 그 사이의 프롬프트 전달과 도구 호출을 측정한다. BM25[3]는 정확한 토큰 매칭에 강하고, SPLADE와 SPLADE++[4][5]는 질의 어휘를 확장해 어휘 불일치를 줄인다. 밀집 검색[6]-[8]은 의미 유사도로 표현이 다른 문서를 찾고, RRF[9]는 점수 체계가 다른 검색 결과를 순위로 결합한다. 이들은 검색 방법과 문서 순위 개선을 다루지만, 제안한 지식 검색 계층은 새 검색기를 제시하는 대신 질의 유형을 미리 알 수 없는 모의해킹 환경에 맞춰 SPLADE++, 밀집 검색, 백링크 그래프를 결합한다. 평가는 평균 순위와 함께 질의 유형별 성능 저하가 완화되는지를 비교한다. 그래프 기반 RAG 연구[10][11]는 문서나 개체의 연결 관계를 검색과 답변 생성에 활용한다. 제안한 백링크 그래프는 독립 검색기가 아니라 희소 검색 상위 결과에서 출발해 연결된 공격 기법을 찾는 보조 경로이며, 효과는 정답 한 건의 순위가 아니라 복합 침투 절차에 필요한 여러 기법의 노트가 검색 결과에 포함된 정도로 측정한다. RAG[2]는 검색된 외부 지식으로 답변을 생성하는 구조를 제시했고, RAGAS[12]는 검색 문서와 답변의 충실도와 관련성을 평가한다. PentestGPT[1]는 하위 과제 분할로 다중 턴 문맥을 관리하며 최종 과제 성공을 평가한다. 그러나 이들만으로는 지식이 프롬프트에 들어오지 않아 실패했는지, 들어왔지만 올바른 도구를 호출하지 못해 실패했는지 구분하기 어렵다. 제안한 평가 체계는 전자를 전달 정확도로, 후자를 행동 충실도로 측정한다.

표 1은 검토한 연구와 본 연구의 평가 단위를 비교한 것이다.

표 1. 관련 연구 접근의 비교
Table 1. Related work comparison

Approach	Measurement unit	Vocabulary mismatch	Prompt delivery	Action linkage
Lexical retrieval [3]	Document rank	Weak	Out of scope	Out of scope
Learned sparse retrieval [4][5]	Document rank	Strong	Out of scope	Out of scope
Dense retrieval [6][7]	Document rank	Strong	Out of scope	Out of scope
Graph-augmented retrieval [10][11]	Document rank	Moderate	Out of scope	Out of scope
RAG evaluation [12]	Answer faithfulness	Not applicable	Not measured	Out of scope
Pentesting agent [1]	Final task success	Not applicable	Not measured	Aggregate only
This work	Prompt and tool call	Strong	Delivery accuracy	Action fidelity

II. 문제 정의

본 연구는 에이전트가 다음 행동을 결정하는 과정을 턴별로 분석한다. 각 턴에서 에이전트가 현재까지 확인한 정보로 상황 질의를 만들고, 지식 검색 계층이 관련 노트 중 상위 4건을 프롬프트에 넣으며, 메인 에이전트는 이를 바탕으로 다음 도구를 선택해 실행한다. 전체 흐름은 검색, 전달, 행동의 세 단계로 나뉜다. 검색 단계는 필요한 노트가 후보 목록에 들어왔는지, 전달 단계는 해당 노트가 상위 4건에 포함되어 프롬프트에 제공되었는지, 행동 단계는 에이전트가 올바른 도구를 호출했는지를 다룬다. 따라서 필요한 노트를 찾지 못했거나 상위 4건에 넣지 못한 경우는 지식 검색 계층의 문제이고, 노트가 제공되었는데도 올바른 도구를 호출하지 못한 경우는 전달 이후의 행동 문제이다. 전달 정확도는 필요한 지식이 상위 4건에 포함된 정도를 나타내고, 행동 충실도는 올바른 도구를 호출했는지와 호출까지 몇 턴이 걸렸는지를 함께 반영한다. 두 지표의 차이는 지식이 프롬프트에 제공된 뒤에도 도구 호출이 늦어지거나 실행되지 않은 정도를 뜻하며, 각 지표의 측정 범위는 그림 1과 같다.

그림 1은 한 턴에서 지식 검색부터 도구 호출까지 이어지는 흐름과 두 지표의 측정 범위를 보여준다. 구체적인 계산 방법은 4.1절에서 설명한다.

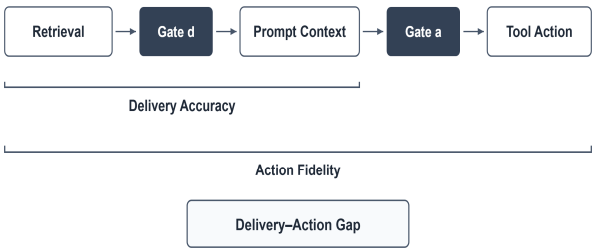


그림 1. 전달 정확도와 행동 충실도의 측정 범위
Fig. 1. Measurement scope of delivery accuracy and action fidelity

III. 동적 컨텍스트 구성 아키텍처

3.1 컨텍스트 구성의 정의와 처리 흐름

컨텍스트 구성은 에이전트가 다음 도구를 선택하기 전에 현재 상황에 맞는 지식을 찾아 프롬프트에 넣는 과정이다. 에이전트는 정찰 결과와 직전 도구의 실행 결과로 검색 질의를 만들며, 새로운 단서가 나오면 질의도 달라지므로 노트를 매 턴 다시 찾는다.

그림 2와 같이 같은 질의를 희소 검색과 밀집 검색에 보낸다. 희소 검색은 명령어와 식별자처럼 정확히 일치하는 단어와 관련 어휘를 찾고, 밀집 검색은 표현이 달라도 의미가 비슷한 문서를 찾는다. 백링크 그래프는 희소 검색 상위 5건에서 위키링크를 2-hop까지 따라가 연관 공격 기법을 추가한다.

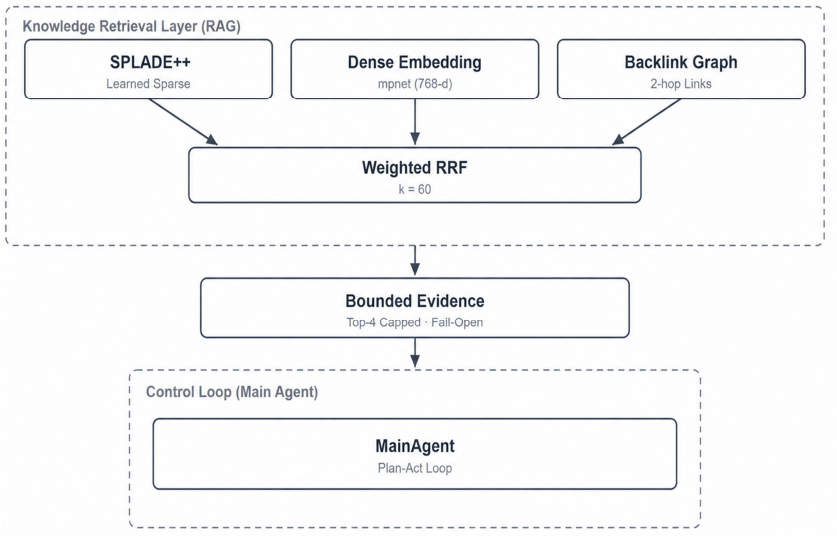


그림 2. 시스템 아키텍처
Fig. 2. System architecture

밀집 검색은 별도로 실행되므로 희소 검색이 놓친 문서도 보완할 수 있다. 세 경로의 문서 순위는 RRF로 합친 뒤 현재 침투 단계와 가까운 문서가 앞에 오도록 다시 정렬하고, 상위 4건만 선택한다. 각 노트는 1,200자 이내로 줄이고 특수 문자를 이스케이프한 뒤, 신뢰할 수 없는 외부 근거라는 표시와 함께 프롬프트에 넣는다[13].

실험에서는 1,197개 어휘 항목을 사용하는 BM25[3]를 어휘 검색의 기준선으로 삼았다. 최종 시스템에는 30,522개 어휘를 쓰는 Splade_PP_en_v2 기반 SPLADE++[4][5]를 희소 검색으로, 문장을 768차원 벡터로 바꾸어 비교하는 all-mpnet-base-v2[6]-[8]를 밀집 검색으로 적용하였다.

세 검색 경로는 문서의 관련성을 판단하는 방식이 달라 점수를 그대로 비교할 수 없으므로, 각 경로가 매긴 순위만 사용하고 높은 순위에 더 큰 값을 주는 가중 RRF[9]로 결과를 합쳤다. 계산식과 가중치는 3.3절에서 설명한다.

지식 검색 계층은 메인 에이전트와 분리해 실행한다. 검색 중 오류가 발생하거나 정해진 시간을 넘기면 해당 턴에는 검색 결과를 전달하지 않고 메인 에이전트가 기존 작업을 계속하며, 이처럼 검색 계층의 문제가 전체 작업을 중단시키지 않는 정책을 장애 허용(Fail-open)이라고 한다.

그림 3은 GraphQL 취약점 진단을 예로 들어, 상황 질의가 세 검색 경로를 거쳐 상위 4건의 노트로 정리되고 도구 호출에 활용되는 과정을 보여준다.

3.2 지식 검색 계층의 설계 과정

지식 검색 계층은 에이전트를 실행하면서 확인한 검색 문제를 하나씩 보완하는 과정에서 발전하였다. 이 과정에서는 메인 에이전트의 동작 방식은 바꾸지 않고, 필요한 지식을 찾아 프롬프트에 넣는 검색 계층만 설계하였다. 설계 과정은 다음 다섯 단계로 정리된다.

1단계: 텍스트 그래프의 한계. 초기 지식 베이스는 위키링크 기반 텍스트 그래프 탐색을 사용하였다[10][11]. 그래프는 이미 찾은 노드에서 연관 기법으로 이동하는 데에는 유용하지만, 자연어 질의만으로 출발 노드를 정하기는 어렵다. 이 때문에 관련성이 낮은 이웃 노드까지 함께 탐색되어 검색 잡음이 증가하였다.

2단계: BM25 도입과 의미 검색의 부재. 잡음을 줄이고 다양한 기술 질의를 처리하기 위해 BM25 어휘 검색을 도입하였다[3]. BM25는 질의와 문서에 같은 명령어, 옵션, 취약점 식별자가 포함되어 있을 때 관련 문서를 안정적으로 찾았다. 그러나 서로 다른 표현을 사용한 질의에는 취약하여 어휘 불일치 유형의 MRR이 0.402에 머물렀고, 단어가 정확히 일치하지 않아도 관련 문서를 찾을 수 있는 검색 방법이 필요하였다.

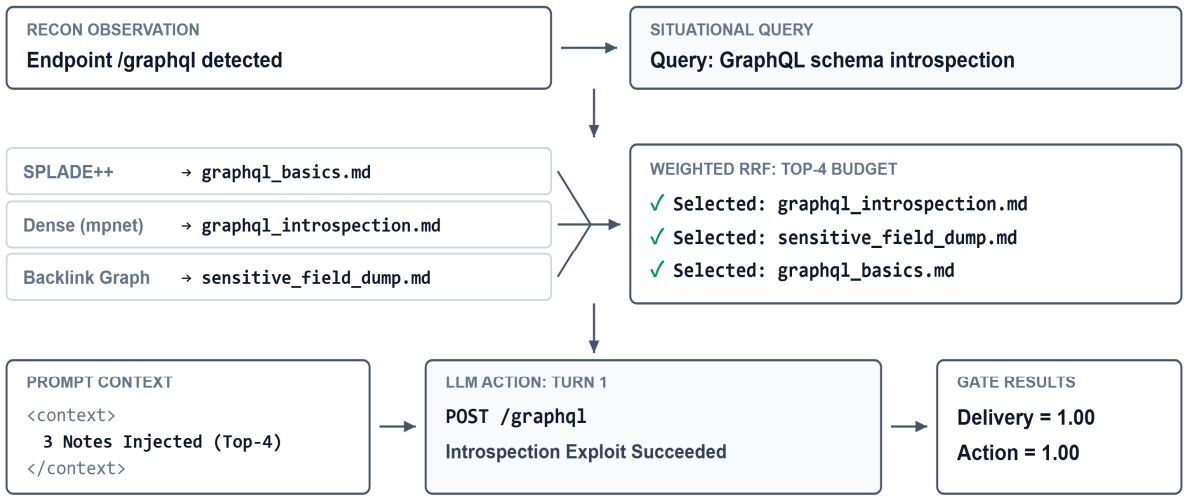


그림 3. 컨텍스트 조립 파이프라인
Fig. 3. Context assembly pipeline

3단계: 밀집 검색 도입과 상호보완적 결합. 공통 토큰이 없는 질의를 찾기 위해 문장 임베딩(all-mpnet-base-v2) 기반 밀집 검색을 도입하였다 [6]-[8]. 밀집 검색은 개념·의역 질의에서 MRR 0.667을 기록했지만, 고유 해시나 명령어 식별자 검색에서는 0.716에 그쳤다. 희소 검색과 밀집 검색은 서로 잘 찾는 질의가 달랐으므로 두 경로를 함께 사용하였다.

4단계: RRF 기반 다중 검색 결합과 그래프 경로 재배치. 어휘 검색과 밀집 검색은 서로 다른 방식으로 점수를 계산하므로, 두 점수를 그대로 합치기 어렵다. 이에 점수 대신 각 검색 결과의 순위를 기준으로 결합하는 가중 상호 순위 융합(RRF)을 도입하였다 [9]. 그래프 검색은 별도의 검색 경로로 사용하지 않고, 어휘 검색 상위 5개 문서에서 출발해 위키링크를 2-hop까지 따라가도록 구성하였다. 이렇게 어휘 검색에서 찾은 기법과 연결된 침투 절차까지 검색 범위에 포함하였다.

5단계: BM25를 SPLADE++로 대체. BM25는 질의와 문서에 같은 단어나 식별자가 있어야 관련 문서를 찾기 쉽다. 그래프 검색도 BM25의 상위 결과에서 시작하므로 이러한 영향을 함께 받았다. 이를 줄이기 위해 BM25를 SPLADE++로 대체하였다 [4][5]. SPLADE++는 질의에 직접 나온 단어와 문맥상 관련된 어휘를 함께 검색한다. 이 검색 결과를 그래프 탐색의 출발점으로 사용하며, 밀집 검색은 별도로 실행한다. 표 2는 검색 경로마다 잘 찾는 질의와 놓치기 쉬운 질의를 비교한 것이다. 어느 한 경로만으로 모든 유형을 처리하기 어려워 세 경로의 결과를 결합하였다.

3.3 융합 방식과 경로별 가중치

세 검색 경로의 순위는 식 (1)의 가중 상호 순위 융합(Weighted RRF)으로 합친다. 각 경로에서 높은 순위에 오른 노트일수록 큰 값을 받으며, 여기에 경로별 가중치를 반영한 합이 최종 점수가 된다.

$$\Sigma_{\text{경로}} \frac{\text{가중치}}{60 + \text{순위}} \tag{1}$$

식 (1)은 후보 노트가 희소·밀집·그래프 검색에서 받은 순위를 하나의 점수로 합치는 방법이다. 분자의 가중치는 각 검색 결과를 얼마나 반영할지를 나타내고, 분모의 순위는 해당 노트가 그 검색 결과에서 몇 위에 올랐는지를 뜻한다. 순위 앞의 60은 순위 차이가 최종 점수에 지나치게 크게 반영되지 않도록 조정하는 값으로, RRF의 표준 설정을 사용하였다 [9]. 이 계산을 세 검색 경로에 각각 적용한 뒤 모두 더하면 노트의 최종 융합 점수가 된다. 특정 경로에서 찾지 못한 노트는 그 경로의 점수를 0으로 처리한다. 같은 노트라도 검색 방법에 따라 순위가 달라질 수 있다. RRF는 각 경로에서 높은 순위에 오른 노트에 더 큰 값을 주고, 세 경로에서 받은 값을 더해 최종 순위를 만든다. 검색 방법마다 점수를 계산하는 기준은 다르지만, RRF는 점수 대신 순위를 사용하므로 세 결과를 바로 합칠 수 있다.

세 경로의 가중치는 희소:밀집:그래프를 1:3:1로 정하였다. 그래프 가중치는 1로 두고, 희소 검색과 밀집 검색은 1:1, 1:3, 1:4를 비교하였다. 두 검색 결과를 같은 비율로 반영한 1:1에서는 가장 낮은 질의 유형의 성능이 0.667이었으며, 밀집 검색만 사용한 0.717보다 낮았다. 밀집 검색을 세 배 더 반영한 1:3에서는 이 값이 0.731로 높아져 최종 비율로 선택하였다.

표 2. 검색 경로별 특성
Table 2. Retrieval path characteristics

Path	Signal	Strength	Limitation addressed by others
BM25 [3]	Lexical overlap	Exact tokens, identifiers	Fails without shared tokens
SPLADE++ (Splade_PP_en_v2) [4][5]	Learned sparse	Exact tokens plus contextual expansion	Weaker on rare identifiers
Dense (all-mpnet-base-v2, 768-d) [6]-[8]	Semantic similarity	Paraphrase and synonym queries	Weaker on exact strings
Backlink graph	Link structure	Co-retrieval of related techniques	Depends on sparse-path seeds
RRF [9]	Rank-only fusion	Scale-invariant combination	Requires ranked lists from paths

밀집 검색을 더 많이 반영한 이유는 전체적으로 더 높은 성능을 보였기 때문이다. 다만 밀집 검색은 명령어나 식별자처럼 글자가 정확히 일치해야 하는 검색에 약할 수 있어 희소 검색도 함께 사용하였다. 1:3은 현재 문서 집합에서 정한 비율이므로 다른 환경에서도 같은 결과가 나온다는 뜻은 아니다.

세 검색 결과를 합친 뒤에는 현재 침투 단계와 관련이 높은 노트를 앞쪽으로 다시 정렬한다. 최종 순위에서 상위 4개 노트만 프롬프트에 넣으며, 필요한 노트가 이 4개 안에 있으면 해당 지식이 에이전트에 전달된 것으로 판단한다.

3.4 평가 문서와 질의셋의 구성

검색 방법을 같은 조건에서 비교하고, 세부 설정을 정하는 과정과 최종 성능을 확인하는 과정을 나누기 위해 평가 자료를 세 부분으로 구성하였다. 평가 문서 집합은 모든 검색 방법이 공통으로 검색할 보안 기법 노트 204개로 이루어진다. 구성 선택 질의셋 147개는 사용할 모델과 가중치를 정하는 데 사용하고, 확장 평가 질의셋 14,533개는 선택한 검색 구성이 더 다양한 공격 기법과 표현에서도 비슷한 성능을 보이는지 검증하는 데 사용하였다. 확장 평가 질의셋은 확장용 질의 시드 460개를 패러프레이즈하여 증강하였다.

첫째, 평가 문서 집합은 실제 침투 과제를 분석해 직접 작성한 기법 노트 11개와 검색 난도를 높이기 위해 규칙으로 생성한 방해 노트 193개로 구성하였다. 11개와 193개는 각각 직접 작성한 문서와 규칙으로 생성한 문서의 수이며, 정답과 오답의 수를 뜻하지 않는다. 각 질의가 요구하는 공격 기법을 설명하는 문서 하나를 정답 노트로 지정하고, 나머지 문서는 검색 순위에서 정답 노트와 경쟁하도록 하였다. 따라서 방해 노트도 해당 기법을 묻는 질의에서는 정답 노트가 될 수 있으며, 모든 검색 방법은 같은 평가 문서 집합을 대상으로 비교하였다.

둘째, 구성 선택 질의셋은 질의의 표현 방식에 따라 검색 성능이 어떻게 달라지는지 비교하고, 이를 바탕으로 세부 모델과 가중치를 정하기 위해 만들었다. 실제 침투 과정에서 확인한 공격 기법 45개마다 정식 명칭을 사용한 질의와 공격 상황을 일반적인 문장으로 설명한 질의를 하나씩 작성하여 90개를 마련하였다. 여기에 정답 노트의 핵심 단어를 사용하지 않고 상황만 설명한 질의 39개와 명령어·식별자처럼 정확한 문자열로 작성한 질의 18개를 더해 총 147개로 구성하였다. 표현이 달라도 같은 공격 기법을 묻는 질의에는 같은 정답 노트를 지정했으며, 이 관계는 검색 실험 전에 공개 저장소에 등록하여 임의로 실험 결과를 조작할 수 없도록 증적 등록하였다.

셋째, 확장 평가 질의셋은 더 많은 공격 기법과 표현에서 최종 검색 성능을 확인하기 위해 만들었다. 구성 선택 질의 147개와 방해 노트에서 작성한 추가 질의를 합친 460개 시드를 증강하여 14,533개 질의를 생성하였다. 확장 과정에서는 표현만 바꾸고 의미는 유지했으므로, 새 질의에도 원본과 같은 정답 노트와 질의 유형을 적용하였다. 다만 어휘 불일치와 명령어·식별자 유형은 실제 침투 데이터에 근거한 구성 선택 질의만을 바탕으로 확장하였다.

구성 선택 질의셋에서 세부 모델과 가중치를 정한 뒤, 확장 평가 질의셋으로 최종 검색 성능을 확인하였다. 두 질의셋은 규모와 구성이 다르므로 MRR을 직접 비교하지 않고, 각 질의셋 안에서 검색 방법 간 차이만 비교하였다. 또한 이번 실험은 모든 검색 방법에 동일한 평가 문서 집합을 사용했으므로, 문서 작성·생성 방식에 따른 성능 차이 분석 연구는 향후 과제로 남긴다.

3.5 검색 구성 비교와 신호 간 상호보완성

평가 문서 집합 204개와 확장 평가 질의셋 14,533개를 사용하여 BM25, SPLADE++, 밀집 검색, 희소·밀집 결합을 비교하였다(표 3). 평가 지표인 MRR@10은 정답 노트가 검색 결과 상위 10건에서 얼마나 앞에 나타나는지를 측정하며, 1에 가까울수록 정답을 높은 순위에서 찾았다는 뜻이다. 연관 기법까지 함께 찾는 그래프 경로는 역할이 다르므로 3.7절에서 별도로 평가하였다.

표 3. 질의 유형별 검색 MRR
Table 3. Retrieval MRR by query type

Configuration	Exact	General	Vocabulary mismatch	Token/ID	Overall	Floor
BM25	0.983	0.934	0.402	0.898	0.884	0.402
SPLADE++ (Splade_PP_en_v2)	0.970	0.932	0.410	0.749	0.880	0.410
Dense (all-mpnet-base-v2, 768-d)	0.976	0.915	0.667	0.716	0.898	0.667
SPLADE++ Dense (all-mpnet-base-v2, 1:3 weighted)	0.978	0.924	0.630	0.770	0.901	0.630

질의는 표현 방식에 따라 네 유형으로 나누었다. 정확(Exact) 유형은 정답 노트와 같은 기술 용어를 사용하고, 일반(General) 유형은 공격 상황을 평이한 문장으로 묻는다. 어휘 불일치(Vocabulary mismatch) 유형은 같은 공격 기법을 다른 단어로 표현하며, 토큰(Token/ID) 유형은 해시나 명령어 식별자처럼 정확히 일치해야 하는 문자열을 사용한다. 질의 수는 정확 2,764개, 일반 10,059개, 어휘 불일치 1,620개, 토큰 90개이다.

표 3의 전체(Overall)는 14,533개 질의 전체에서 계산한 평균이므로 질의 수가 많은 유형의 영향이 더 크게 반영된다. 하한 성능(Floor)은 네 유형 가운데 각 검색 구성의 MRR이 가장 낮았던 값을 나타낸다. 표 3에서는 모든 질의 유형에서 가장 높은 성능을 보인 단일 검색기가 없었다. BM25는 정답 노트와 같은 용어를 사용한 정확 유형에서 0.983, 해시와 식별자를 사용한 토큰 유형에서 0.898을 기록했지만, 다른 단어로 표현한 어휘 불일치 유형에서는 0.402로 낮아졌다. 반면 밀집 검색은 어휘 불일치 유형에서 0.667로 가장 높았으나 토큰 유형에서는 0.716을 기록하였다. 이처럼 강점을 보이는 질의 유형이 달랐기 때문에 SPLADE++ 희소 검색과 밀집 검색을 결합하였다.

그림 4는 검색 방법마다 강한 질의 유형과 약한 질의 유형이 다름을 보여준다. BM25, SPLADE++, 밀집 검색은 각각 적어도 한 유형에서 네 구성 중 가장 낮은 MRR을 기록했지만, SPLADE++와 밀집 검색을 결합한 구성은 어느 유형에서도 최하위가 아니었다. 그러나 각 구성에서 가장 낮은 MRR만 비교하면 밀집 검색 단독이 0.667로 결합 구성의 0.630보다 높아, 결합 구성이 모든 지표를 개선한 것은 아니다. 대신 밀집 검색이 약했던 토큰 유형의 MRR을 0.716에서 0.770으로 보완하면서 전체 MRR 0.901을 기록하였다. 이는 질의의 표현 방식을 미리 알기 어려울 때 에이전트의 검색이 놓치는 유형을 다른 방법이 보완할 수 있음을 보여주며, 침투 과제 평가에서도 같은 경향이 나타났다. 개별 질의 유형의 검색 정확도도 중요하지만, 자율 모의해킹 과정에서는 다음 턴에 명령어·식별자가 들어올지, 같은 기법을 다른 말로 표현한 질의가 들어올지 미리 알기 어렵다. 따라서 희소 경로에는 정확한 문자열을 찾으면서도 관련 어휘까지 검색 범위를 넓힐 수 있는 유연한 검색 방식이 필요하다. SPLADE++는 이러한 어휘 확장 능력과 검색 성능이 선행 연구에서 보고되었으므로 [4][5], 다양한 표현에 대응하기 위한 최종 희소 검색으로 선택하였다. 다만 현재 실험에서는 어휘 불일치 MRR이 SPLADE++ 0.410과 BM25 0.402로 비슷했고, 토큰 유형에서는 BM25가 0.898로 SPLADE++의 0.749보다 높았으며 전체 평균도 큰 차이가 없었다. 따라서 이번 선택은 현재 평가 문서 집합에서의 절대적인 성능 우위보다, 어떤 형태의 질의가 들어와도 검색 범위를 유연하게 넓힐 수 있다는 점에 집중한다.

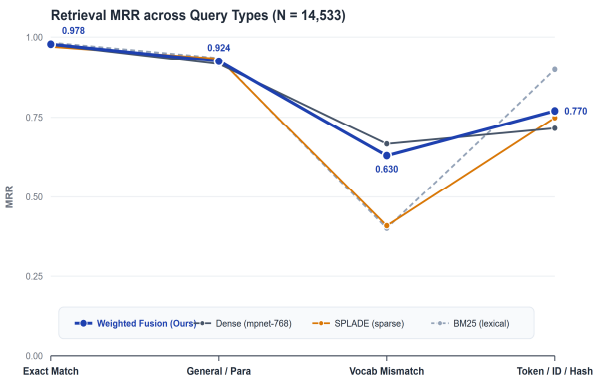


그림 4. 질의 유형별 MRR
Fig. 4. MRR by query type

3.6 구성요소의 선택

검색 경로와 결과를 합치는 순서를 먼저 정한

뒤, 구성 선택 질의셋 147개로 여러 후보를 비교하였다. 그 결과를 바탕으로 밀집 임베딩 모델, SPLADE++ 세부 모델, 그래프 탐색 범위, 검색 경로별 가중치를 선택하였다(표 4~7).

표 4는 같은 공격 기법을 다른 단어로 표현했을 때 각 밀집 임베딩 모델이 정답 노트를 얼마나 잘 찾는지 비교한 결과이다. all-mpnet-base-v2는 어휘 불일치 유형에서 0.764, 전체에서 0.856을 기록했으며, 가장 낮은 유형의 성능도 0.721로 다른 384차원 모델보다 높았다. 명령어와 식별자를 찾는 토큰 유형에서는 all-MiniLM-L6-v2가 0.732로 더 높았다. 최종 검색 구조는 희소 검색으로 명령어와 식별자를 찾고, 밀집 검색으로 표현이 다른 질의를 보완하는 구조이므로, 밀집 모델은 토큰 유형보다 어휘 불일치 유형의 성능을 우선하여 선택했으며, 이 유형에서 가장 높은 all-mpnet-base-v2를 사용하였다.

표 4. 밀집 임베딩 모델 비교
Table 4. Dense embedding models

Model	Dim	Vocabulary mismatch	Token/ID	Overall	Floor
all-MiniLM-L6-v2 [14]	384	0.633	0.732	0.816	0.633
bge-small-en-v1.5 [15]	384	0.639	0.602	0.806	0.602
gte-small [16]	384	0.632	0.681	0.828	0.632
e5-small-v2 [17]	384	0.597	0.662	0.805	0.597
all-mpnet-base-v2 [8]	768	0.764	0.721	0.856	0.721

표 5는 SPLADE++의 두 버전을 비교한 결과이다. Splade_PP_en_v2는 성능이 높은 모델의 판단을 학습하고, 정답과 비슷해 혼동하기 쉬운 오답 문서를 구별하는 학습 방식을 v1보다 개선한 모델이다. 어휘 불일치 유형에서는 v1의 0.486보다 높은 0.519를 기록했고, 전체 MRR도 0.769에서 0.790으로 높아져 최종 모델로 선택하였다. 이후 본문에서 SPLADE++는 Splade_PP_en_v2를 뜻한다. 표 3은 14,533개 확장 평가 질의셋, 표 5는 147개 구성 선택 질의셋에서 얻은 결과이므로 두 표의 MRR 값은 직접 비교하지 않는다.

표 5. SPLADE++ 체크포인트 비교
Table 5. SPLADE++ checkpoints

Checkpoint	Exact	General	Vocabulary mismatch	Token/ID	Overall	Floor
Splade_PP_en_v2	0.978	0.854	0.519	0.749	0.790	0.519
Splade_PP_en_v1	0.956	0.820	0.486	0.784	0.769	0.486

표 6은 백링크 그래프가 문서의 링크를 몇 단계까지 따라가야 하는지 비교한 결과이다. 링크를 한 번 따라가는 범위를 1-hop, 두 번 따라가는 범위를 2-hop이라고 한다. 같은 범주의 공격 기법을 찾은 비율은 1-hop에서 55.6%였고, 2-hop으로 넓히자 59.5%로 높아졌다. 그러나 3-hop까지 넓혀도 이 비율은 59.5%로 같았고, 검색 후보만 평균 4.13개에서 4.16개로 늘어났다. 따라서 연관 기법을 더 찾는 효과가 나타나면서 불필요한 후보 증가는 적은 2-hop을 선택하였다.

표 6. 그래프 확장 홉 수 비교
Table 6. Graph expansion hops

Hop	Mean expanded notes	Same-category coverage
1	3.65	0.556
2	4.13	0.595
3	4.16	0.595

표 7은 희소 검색과 밀집 검색의 결과를 합칠 때 어느 쪽에 더 큰 비중을 둘지 비교한 결과이다. 1:1은 두 검색 결과를 같은 비중으로 반영하고, 1:3은 밀집 검색을 희소 검색보다 세 배 크게 반영한다. 두 경로를 1:1로 합쳤을 때 전체 MRR은 0.836이었고, 네 질의 유형 가운데 가장 낮은 MRR은 0.667이었다. 비율을 1:3으로 바꾸자 두 값이 각각 0.865와 0.731로 높아졌지만, 밀집 검색의 비중을 1:4까지 높여도 더 나아지지 않았다. 이에 구성 선택 질의셋에서 전체 성능과 가장 낮은 유형의 성능이 함께 높았던 1:3을 선택하였다. 표 4와 표 7은 별도로 측정하여 밀집 검색의 단독 점수에 소수점 셋째 자리의 차이가 있으므로, 가중치에 따른 차이는 표 7 안에서 비교하였다.

표 7. RRF 융합 가중치 비교
Table 7. RRF fusion weights

Weight	Vocabulary mismatch	Token/ID	Overall	Floor
Dense only (all-mpnet-base-v2)	0.764	0.717	0.855	0.717
1:1 (equal)	0.667	0.739	0.836	0.667
1:3	0.763	0.731	0.865	0.731
1:4	0.756	0.731	0.863	0.731

확장 평가 질의셋에서도 1:3의 가장 낮은 유형 점수는 0.630으로, 1:1의 0.590보다 높았다. 다만 밀집 검색만 사용한 0.667보다는 낮아, 두 경로를 합친 구성이 모든 지표에서 우수한 것은 아니었다. 1:3은 한 유형에만 맞추기보다 어떤 형태의 질의가 들어올지 모르는 상황에서 두 검색 방법을 함께 활용하기 위해 선택한 비율이다. 밀집 검색 쪽에 큰 가중치를 둔 구성이 BM25 단독이나 FAISS 단독보다 높은 검색 성능을 보인 사례는 국내 학술지에서도 보고되었다[18]. 다만 평가 문서와 지표가 달라 결과를 직접 비교하지는 않는다.

3.7 그래프의 연관 근거 커버리지

앞선 실험은 질의마다 정답 노트 하나를 정하고, 검색기가 이 노트를 몇 위에서 찾는지 평가하였다. 그래프 검색은 중심이 되는 공격 기법뿐 아니라, 그 공격을 수행할 때 함께 필요한 보조 기법까지 찾기 위해 사용한다. 이 기능을 확인하기 위해 중심 기법을 묻는 질의 6개를 준비하고, 각 질의와 함께 검색되어야 할 보조 기법 노트를 미리 정하였다. 보조 기법 노트는 모두 12개이며, 이 가운데 몇 개가 검색 결과의 상위 4건과 10건에 포함되는지 측정하였다(표 8).

그래프는 보조 기법을 검색 후보에 추가하는 데 효과가 있었지만, 프롬프트에 넣는 노트가 상위 4건으로 제한되어 실제 전달 비율의 증가 폭은 작았다. 그래프를 사용하지 않았을 때는 필요한 보조 기법 노트의 47.2%가 검색 결과 상위 10건에 포함되었고, 그래프를 사용했을 때는 86.1%가 포함되었다. 즉, 그래프를 추가하면서 더 많은 보조 기법을 후보로 찾을 수 있었다.

표 8. 그래프 연관 근거 커버리지
Table 8. Graph evidence coverage

Metric	Graph OFF	Graph ON
Related-note Recall@4	0.222	0.306
Related-note Recall@10	0.472	0.861

그래프가 추가한 후보도 RRF 통합과 재정렬을 거쳐야 하며, 최종 순위 중 상위 4건만 프롬프트에 들어간다. 이 4건에 포함된 보조 기법 노트의 비율은 그래프를 사용하지 않았을 때 22.2%, 사용했을 때 30.6%였다(표 8, 그림 5). 그래프가 보조 기법을 찾아도 상당수가 5위 이하에 머물러 실제 전달로 이어지지 않은 것이다. 즉, 문제는 필요한 기법을 찾는 데만 있지 않고, 찾은 기법 가운데 무엇을 프롬프트의 제한된 네 자리에 넣을지에도 있었다. 따라서 복합 침투 과제에 필요한 기법을 상위 4건으로 올릴 수 있도록 재정렬 기준을 개선하고 그 효과를 검증하는 작업은 향후 연구 과제로 남긴다.

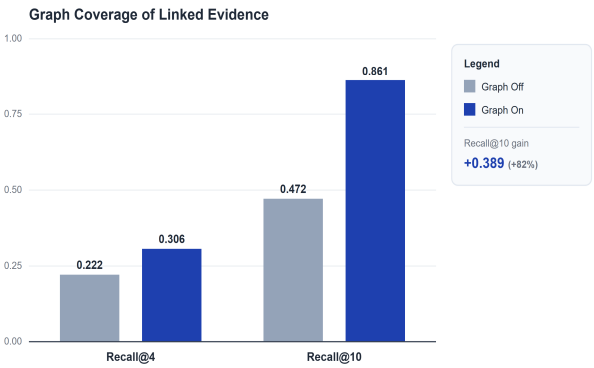


그림 5. 그래프 연관 근거 커버리지
Fig. 5. Graph evidence coverage

IV. 평가 설계 및 실측

4.1 전달 정확도와 행동 충실도

평가는 필요한 지식이 에이전트에 전달되었는지와, 전달된 지식이 실제 도구 호출로 이어졌는지를 나누어 측정한다. 먼저 전달 정확도(DA, Delivery Accuracy)는 침투 상황에 필요한 정답 지식이 프롬프트 상위 4건에 얼마나 포함되었는지를 나타내며, 식 (2)로 계산하였다.

$$\frac{\Sigma \text{중요도} \times \text{일치도} \times \text{전달}}{\Sigma \text{중요도}} \quad (2)$$

식 (2)의 합계 기호(Σ)는 과제에 필요한 모든 정답 지식의 값을 더한다는 뜻이다. 분자에서는 정답 지식마다 중요도, 일치도, 전달 값을 곱한 뒤 합산한다. 중요도는 해당 지식이 과제에서 차지하는 비중이며, 일치도는 검색된 노트가 미리 정한 정답 노트와 같은지를 나타낸다. 본 평가에서는 검증된 정답 노트만 사용하므로 일치도는 1로 고정하였다. 전달은 정답 노트가 프롬프트 상위 4건에 포함되면 1, 포함되지 않으면 0이다. 분자의 합계를 모든 정답 지식의 중요도 합계로 나누면 전달 정확도가 되며, 값이 1이면 필요한 지식이 모두 프롬프트에 포함되었다는 뜻이다.

다음으로 행동 충실도(AF, Action Fidelity)는 전달된 지식이 올바른 도구 호출로 이어졌는지와 호출까지 몇 턴이 걸렸는지를 함께 반영하며, 식 (3)으로 계산한다.

$$\frac{\Sigma \text{중요도} \times \text{일치도} \times \text{전달} \times \text{행동}}{\Sigma \text{중요도}} \quad (3)$$

식 (3)은 식 (2)의 중요도, 일치도, 전달 값에 행동 점수를 하나 더 곱한다. 식에서는 행동 점수를 행동으로 줄여 표기한다. 행동 점수는 전달된 지식과 관련된 도구를 호출했는지와 호출까지 몇 턴이 걸렸는지를 나타낸다. 정답 노트가 프롬프트에 포함되지 않으면 전달 값이 0이 되어 행동 점수도 계산 결과에 반영되지 않는다. 정답 노트가 포함된 경우에는 행동 점수까지 곱해 합산하고, 이를 모든 정답 지식의 중요도 합계로 나누어 행동 충실도를 구한다. 값이 1에 가까울수록 필요한 도구를 빠짐없이 이른 시점에 호출했다는 뜻이다. 행동 점수는 도구 호출 여부와 최초 호출 시점을 반영하여 식 (4)로 계산한다.

$$\text{호출} \times \text{최대}(0,\ 1 - 0.1(\text{시점} - 1)) \quad (4)$$

식 (4)의 호출은 필요한 도구를 호출하면 1, 끝까지 호출하지 않으면 0이다. 시점은 해당 도구를 처음 호출한 턴의 번호를 뜻한다. 첫 번째 턴에 호출하면 행동 점수는 1이며, 호출이 한 턴 늦어질 때마다 0.1씩 줄어 네 번째 턴에는 0.7이 된다. 최대는 두 값 중 큰 값을 고르는 함수로 계산 결과가 0보다 작아지는 것을 막고, 도구를 끝까지 호출하지 않으면 호출이 0이므로 행동 점수도 0이 된다. 호출 시점이 늦을수록 점수를 줄인 이유는 그동안 추가 탐색에 프롬프트 예산과 추론 비용이 발생하기 때문이다.

두 지표의 차이를 전달-행동 격차(GAP)로 정의하며, 계산식은 식 (5)와 같다.

$$\text{전달 정확도} - \text{행동 충실도} \geq 0 \quad (5)$$

전달-행동 격차는 지식이 프롬프트에 들어간 뒤에도 올바른 도구 호출로 이어지지 않은 정도를 나타낸다. 식 (5)와 같이 전달 정확도에서 행동 충실도를 빼서 계산한다. 전달된 지식을 모두 첫 번째 턴에 사용하면 두 점수가 같아 격차는 0이 되지만, 도구 호출이 늦어지거나 끝까지 이루어지지 않으면 행동 충실도가 낮아져 격차가 커진다. 행동 충실도는 0부터 1 사이의 행동 점수를 반영하므로 전달 정확도보다 높아질 수 없으며, 이에 따라 격차도 항상 0 이상이다.

행동 점수는 자동 판정과 저자 확인의 두 단계를 거쳐 확정하였다. 먼저 언어 모델 기반 판정기가 실행 로그를 읽고 도구를 호출했는지와 몇 번째 턴에 호출했는지를 판정하였다. 이후 저자가 원시 로그의 HTTP 요청과 공격 페이로드를 직접 확인하여 자동 판정 결과와 대조하였다. 5개 과제에 필요한 지식 7개 항목에서는 두 판정 결과가 같은 것으로 관측되었다. 코헨 카파 계수는 우연히 같은 판정이 나올 가능성을 제외하고 두 판정이 얼마나 일치하는지를 나타내며, 1에 가까울수록 일치도가 높다. 이번 표본에서는 1.00으로 측정되었지만, 제한된 항목에서 얻은 결과이므로 다른 과제에서도 동일한 수치가 나타난다고 단정할 수는 없다.

4.2 전달-행동 격차의 실측

XBEN 검증 과제군 [19] 가운데 OWASP Top 10을 반영한 대표 과제 12개를 대상으로, 검색된 지식

이 프롬프트에 들어간 뒤 실제 도구 호출로 이어지는지를 확인하였다. 검색 대상은 앞선 실험과 같은 보안 노트 204개이다. 전달된 지식과 이후의 행동을 비교하려면 과제에 필요한 지식이 먼저 프롬프트에 포함되어야 한다. 12개 과제 가운데 이 조건을 충족한 5개 과제를 대상으로 전달 정확도와 행동 충실도를 비교하였다(표 9).

5개 과제는 모두 필요한 지식이 프롬프트에 포함되어 전달 정확도 1.00을 기록하였다. 그러나 올바른 도구를 호출했는지와 얼마나 빨리 호출했는지를 반영한 행동 충실도는 평균 0.58이었다. 두 점수 사이의 차이인 0.42 가운데 0.30은 필요한 도구를 끝내 호출하지 않아 발생했고, 0.12는 호출이 늦어지면서 발생하였다. 필요한 지식이 이미 제공된 뒤에 생긴 차이이므로 검색 과정만으로는 이 결과를 설명하기 어렵다.

과제별 행동 점수는 도구 호출 시점과 실행 여부에 따라 달랐다. XBEN-006은 지식을 전달받은 첫 턴에 공격 도구를 실행하여 1.00을 기록하였다. XBEN-009와 XBEN-057은 정찰을 거쳐 4번째 턴에 도구를 호출하여 0.70을 기록하였다. XBEN-005는 IDOR 기법을 사용했지만 JWT 권한 상승 공격으로 이어지지 않아 0.50을 기록하였다. XBEN-020은 필수 지식이 검색 1위로 제공되었지만 정찰 명령을 반복하고 공격 도구를 호출하지 않아 0.00을 기록하였다.

그림 6에서 5개 과제의 전달 정확도는 모두 1.00이었으나 평균 행동 충실도는 0.58로, 두 지표 사이에 평균 0.42의 차이가 관측되었다. 이는 전달된 지식을 도구 호출로 옮기는 과정의 인지 병목 가능성을 시사하지만, 단일 모델과 제한된 과제의 결과이므로 여러 모델과 과제에서 검증하는 작업은 향후 연구 과제로 남긴다.

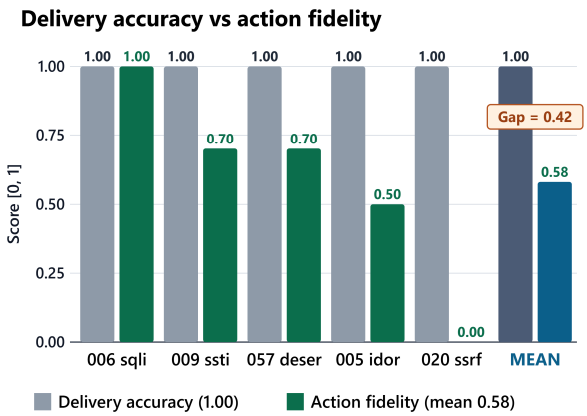


그림 6. 전달 정확도와 행동 충실도
Fig. 6. Delivery accuracy vs. action fidelity

4.3 검색 경로 조합별 전달 양상

전달 실패가 특정 검색 경로가 없어서 발생하는지 확인하기 위해, 12개 과제를 대상으로 검색 경로의 4가지 활성화 조합을 비교 평가하였다.

그림 7에서 기법 명칭이 질의에 직접 포함된 과제는 검색 경로의 조합과 관계없이 정답 지식이 1위에 올랐다. 그러나 여러 기법이 필요한 복합 과제에서는 경로별 결과가 달랐는데, 역직렬화 공격은 SPLADE++만 찾았고 요청 스머글링 공격은 all-mpnet-base-v2만 찾았다. 이처럼 두 경로가 서로 놓친 과제를 보완하는 경향이 관측되었다. 반면 어떤 경로를 조합해도 정답 지식을 찾지 못한 과제는 초기 단서가 부족하여 상황 질의에 필요한 공격 기법이 반영되지 않았다. 이 문제는 검색 경로를 추가하는 것만으로 해결하기 어려우므로, 침투 진행 상태에 맞춰 상황 질의를 다시 만드는 방법은 향후 연구 과제로 남긴다.

표 9. 전달 정확도와 행동 충실도
Table 9. Delivery accuracy and action fidelity

Task	Vulnerability	Delivery accuracy	Action gate	Observed behaviour
XBEN-006	SQL injection	1.00	1.00	Tool executed at turn 1
XBEN-009	SSTI	1.00	0.70	Correct tool at turn 4 after recon
XBEN-057	Deserialization	1.00	0.70	Correct tool at turn 4 after recon
XBEN-005	IDOR / JWT	1.00	0.50	IDOR used, JWT escalation never attempted
XBEN-020	SSRF	1.00	0.00	Evidence ranked first, recon only, no tool call
Mean		1.00	0.58	Delivery–action gap 0.42



그림 7. 검색 경로 조합별 전달률
Fig. 7. Delivery rate by path combination

V. 논 의

5.1 본 연구의 독창성과 기여

기존 연구는 주로 정답 문서의 순위나 최종 과제의 성공 여부를 평가하였다. 본 연구는 검색된 지식이 프롬프트에 포함되고 실제 도구 호출로 이어지는 과정을 나누어 측정함으로써, 실패가 지식 전달 전후의 어느 단계에서 발생했는지 구분하였다.

첫째, 희소 검색과 밀집 검색을 결합하여 서로 다른 표현의 질의를 함께 처리하였다. 두 검색 방법은 잘 찾는 질의 유형이 달랐으며, 결합 구성에서는 한 경로가 놓친 정답 지식을 다른 경로가 찾는 상호보완 관계가 관측되었다.

둘째, 백링크 그래프의 효과를 정답 노트 한 건의 순위가 아니라 함께 필요한 보조 기법을 얼마나 찾았는지로 평가하였다. 이를 통해 그래프가 검색 후보를 넓히는 효과와, 찾은 노트가 실제 프롬프트에 들어가는 효과를 구분하였다.

셋째, 전달 정확도와 행동 충실도를 따로 측정하여 지식이 전달된 뒤에도 도구 호출이 늦거나 이루어지지 않는 경우를 확인하였다. 필요한 지식이 모두 전달된 5개 과제에서 두 지표 사이에 평균 0.42의 차이가 관측되었으며, 이는 검색 이후의 행동 과정도 함께 살펴볼 필요가 있음을 시사한다.

5.2 적용 조건과 한계

이번 평가는 연구에서 구성한 문서와 질의, 제한된 수의 과제, 하나의 기반 모델을 사용했으므로 결과는 현재 실험 범위 안에서 해석해야 한다. 검색 경로의 가중치와 평가 기준도 이번 실험에 맞춰 정했기 때문에 다른 환경에서 같은 결과가 나타나는지는 확인하지 못했다. 따라서 이번 결과는 특정 구성의 일반적인 우수성보다 검색·전달·행동 단계에서 어떤 차이가 나타났는지를 보여주는 관측으로 해석한다. 다른 데이터와 모델을 사용한 추가 검증은 후속 연구에서 다룬다.

VI. 결론 및 향후 연구 과제

본 논문은 자율 모의해킹 에이전트가 현재 상황에 필요한 지식을 선택해 프롬프트에 넣는 과정을 컨텍스트 구성 문제로 보았다. 이를 위해 서로 다른 검색 신호를 결합한 지식 검색 계층을 구성하고, 지식 전달과 도구 호출을 구분하여 살펴보았다.

실험에서는 희소 검색과 밀집 검색이 서로 다른 질의에서 놓친 지식을 보완하는 경향이 관측되었고, 그래프 검색을 사용했을 때 함께 필요한 기법이 검색 결과에 더 많이 포함되었다. 필요한 지식이 프롬프트에 제공된 뒤에도 올바른 도구 호출로 이어지지 않는 경우가 나타나, 검색 결과와 이후의 행동을 함께 살펴볼 필요가 있음을 시사한다.

향후에는 다양한 침투 데이터와 기반 모델에서 같은 경향이 나타나는지 추가로 확인할 필요가 있다. 복합 과제의 진행 상황에 맞춰 필요한 지식을 구성하는 방법도 후속 연구에서 살펴보고자 한다.

References

[1] G. Deng, Y. Liu, V. Mayoral-Vilches, P. Liu, Y. Li, Y. Xu, T. Zhang, Y. Liu, M. Pinzger, and S. Rass, "PentestGPT: Evaluating and harnessing large language models for automated penetration testing", Proc. of the 33rd USENIX Security Symposium (USENIX Security '24), Philadelphia,

Pennsylvania, USA, pp. 847-864, Aug. 2024.

[2] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Kuttler, M. Lewis, W. Yih, T. Rocktaschel, S. Riedel, and D. Kiela, "Retrieval-augmented generation for knowledge-intensive NLP tasks", Advances in Neural Information Processing Systems, Online, Vol. 33, pp. 9459-9474, Dec. 2020. https://doi.org/10.48550/arXiv.2005.11401.

[3] S. Robertson and H. Zaragoza, "The probabilistic relevance framework: BM25 and beyond", Foundations and Trends in Information Retrieval, Vol. 3, No. 4, pp. 333-389, Sep. 2009. https://doi.org/10.1561/1500000019.

[4] T. Formal, B. Piwowarski, and S. Clinchant, "SPLADE: Sparse lexical and expansion model for first stage ranking", Proc. of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Online, pp. 2288-2292, Jul. 2021. https://doi.org/10.1145/3404835.3463098.

[5] T. Formal, C. Lassance, B. Piwowarski, and S. Clinchant, "From distillation to hard negative sampling: Making sparse neural IR models more effective", Proc. of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, pp. 2353-2359, Jul. 2022. https://doi.org/10.1145/3477495.3531857.

[6] V. Karpukhin, B. Oguz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, and W. Yih, "Dense passage retrieval for open-domain question answering", Proc. of the 2020 Conference on Empirical Methods in Natural Language Processing, Online, pp. 6769-6781, Nov. 2020. https://doi.org/10.18653/v1/2020.emnlp-main.550.

[7] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks", Proc. of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, Hong Kong, China, pp. 3980-3990, Nov. 2019. https://doi.org/10.18653/v1/D19-1410.

[8] K. Song, X. Tan, T. Qin, J. Lu, and T. Liu, "MPNet: Masked and permuted pre-training for language understanding", Advances in Neural Information Processing Systems, Online, Vol. 33, pp. 16857-16867, Dec. 2020. https://doi.org/10.48550/arXiv.2004.09297.

[9] G. V. Cormack, C. L. A. Clarke, and S. Buettcher, "Reciprocal rank fusion outperforms Condorcet and individual rank learning methods", Proc. of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval, Boston, Massachusetts, USA, pp. 758-759, Jul. 2009. https://doi.org/10.1145/1571941.1572114.

[10] D. Edge, H. Trinh, N. Cheng, J. Bradley, A. Chao, A. Mody, S. Truitt, D. Metropolitansky, R. O. Ness, and J. Larson, "From local to global: A graph RAG approach to query-focused summarization", arXiv preprint, arXiv:2404.16130, Apr. 2024. https://doi.org/10.48550/arXiv.2404.16130.

[11] X. He, Y. Tian, Y. Sun, N. V. Chawla, T. Laurent, Y. LeCun, X. Bresson, and B. Hooi, "G-Retriever: Retrieval-augmented generation for textual graph understanding and question answering", Advances in Neural Information Processing Systems, Vol. 37, pp. 132876-132907, Dec. 2024. https://doi.org/10.48550/arXiv.2402.07630.

[12] S. Es, J. James, L. Espinosa-Anke, and S. Schockaert, "RAGAs: Automated evaluation of retrieval augmented generation", Proc. of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations, St. Julian's, Malta, pp. 150-158, Mar. 2024. https://doi.org/10.18653/v1/2024.eacl-demo.16.

[13] K. Greshake, S. Abdelnabi, S. Mishra, C. Endres,

T. Holz, and M. Fritz, "Not what you've signed up for: Compromising real-world LLM-integrated applications with indirect prompt injection", Proc. of the 16th ACM Workshop on Artificial Intelligence and Security, Copenhagen, Denmark, pp. 79-90, Nov. 2023. https://doi.org/10.1145/3605764.3623985.

[14] W. Wang, F. Wei, L. Dong, H. Bao, N. Yang, and M. Zhou, "MiniLM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers", Advances in Neural Information Processing Systems, Online, Vol. 33, pp. 5776-5788, Dec. 2020. https://doi.org/10.48550/arXiv.2002.10957.

[15] S. Xiao, Z. Liu, P. Zhang, N. Muennighoff, D. Lian, and J. Nie, "C-Pack: Packed resources for general Chinese embeddings", Proc. of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, Washington, D.C., USA, pp. 641-649, Jul. 2024. https://doi.org/10.1145/3626772.3657878.

[16] Z. Li, X. Zhang, Y. Zhang, D. Long, P. Xie, and M. Zhang, "Towards general text embeddings with multi-stage contrastive learning", arXiv preprint, arXiv:2308.03281, Aug. 2023. https://doi.org/10.48550/arXiv.2308.03281.

[17] L. Wang, N. Yang, X. Huang, B. Jiao, L. Yang, D. Jiang, R. Majumder, and F. Wei, "Text embeddings by weakly-supervised contrastive pre-training", arXiv preprint, arXiv:2212.03533, Dec. 2022. https://doi.org/10.48550/arXiv.2212.03533.

[18] H. S. Ahn, H. S. Lee, and G. W. Kim, "A Study on RAG Retrieval Parameter Optimization using BM25+FAISS Hybrid Retrieval for Aerospace and Defense Technical Documents in a Closed Network Environment", Journal of KIIT, Vol. 24, No. 7, pp. 23-34, Jul. 2026. https://doi.org/10.14801/jkiit.2026.24.7.23.

[19] XBOW Engineering, "XBOW Validation Benchmarks",https://github.com/xbow-engineering/validation-benchmarks. [accessed: Oct. 01, 2026]

저자소개

양 주 성 (Jusung Yang)



2023년 2월 : 광주대학교 정보통신공학과(석사)
2025년 3월 ~ 현재 : 광주대학교 광·정보통신공학과 박사과정
2021년 2월 ~ 현재 : 레피소드 주식회사 대표이사
관심분야 : AI, DevOps

박 상 우 (Sang Woo Park)



2018년 2월 : 광주대학교 음악학과(학사)
2023년 5월 ~ 현재 : 레피소드 주식회사 선임
관심분야 : AI, DevSecOps, Pentest, Embedded

김 광 현 (Gwang-Hyun Kim)



1989년 2월 : 광운대학교 전자계산학과(이학사)
1991년 2월 : 광운대학교 전자계산학과(석사)
1997년 2월 : 광운대학교 전자계산학과(박사)
1997년 3월 ~현재 : 광주대학교 AI소프트웨어학과 교수
관심분야 : 머신러닝/딥러닝, 차세대네트워크 구축