



Received: 2026/08/19
Revised: 2026/08/31
Accepted: 2026/09/12
Published: 2026/09/30

***Corresponding Author:**

Jae-Seong Park

20, Daewangpangyo-ro 851beon-gil, Sujeong-gu,
Seongnam-si, Gyeonggi-do, 13449, Republic of Korea
Tel: +82-31-8038-0592
E-mail: jaeseong.park01@ligdna.com

수명주기관리계획서의 선별적 개정을 위한 LLM-치환규칙 기반 문서 처리 방안 연구

Study on an LLM and Replacement Rule-Based Document Processing Approach for Selective Revision of Life Cycle Sustainment Plan

박재성^{1*}, 이승률², 정준², 오준혁³

¹LIG Defense&Aerospace 미래전장IPS연구소 연구원

²LIG Defense&Aerospace 미래전장IPS연구소 수석연구원

³LIG Defense&Aerospace 미래전장IPS연구소 선임연구원

Jae-Seong Park^{1*}, Seung-Ryool Lee², Jun Jeong², Jun-Hyeok Oh³

^{1,3}Research engineer, Future Warfare Integrated Product Support R&D
(Maritime), LIG Defense&Aerospace

²Chief research engineer, Future Warfare Integrated Product Support R&D
(Maritime), LIG Defense&Aerospace

Abstract

본 논문에서는 기존 사업의 수명주기관리계획서를 신규사업 정보에 맞게 개정하기 위한 LLM 및 치환규칙 기반 문서 처리 방안을 제안한다. 수명주기관리계획서는 사업별 정보가 반복적으로 포함되는 장문 기술문서로 기존 문서 구조를 유지하면서 필요한 항목을 일관되게 수정할 필요가 있다. 그러나 LLM 단독으로 문서를 직접 재작성하는 경우 원문과 관련 없는 내용이 생성되거나 문서 구조가 변형될 가능성이 있다. 이에 본 연구에서는 LLM을 개정 후보 탐색 및 제안 수단으로 활용하고, 사용자가 승인한 치환규칙만 문서에 반영하는 방식을 구성하였다. 기존 문서 구조를 유지하면서 개정 대상 표현을 선별적으로 반영하고, 변경 이력과 잔존 후보를 검토할 수 있는 문서 처리 절차를 제시하였다.

This study proposes a document processing method based on an LLM and replacement rules for revising an existing life cycle sustainment plan using new project information. A life cycle sustainment plan is a long-form technical document that repeatedly contains project-specific information. Therefore, necessary items must be revised consistently while preserving the structure of the existing document. However, when an LLM directly rewrites the document, it may generate content unrelated to the source document or alter the original document structure. To address this issue, this study used an LLM to identify and suggest revision candidates and applied only user-approved replacement rules to the document. The proposed method selectively reflects expressions requiring revision while preserving the existing document structure and supports the review of change history and residual candidates.

Keywords

대규모 언어모델(Large Language Model), 수명주기관리계획서(Life Cycle Sustainment Plan), 치환규칙(Replacement Rule), 사용자 승인(Selective Revision), 문서 처리(Document Processing)

1. 서론

수명주기관리계획서(LCSP, life cycle sustainment plan)는 총 수명주기관리를 목적으로 개발단계에서 획득 및 운영유지 단계까지 적용된다[1]. 따라서 사업 진행 단계와 운용환경 변화에 따라 지속적으로 반영 및 관리해야 하는 문서로 볼 수 있다[2]. 실제로 해군 사업의 수명주기관리계획서를 분석한 결과 사업명, 체계명 등 관련 정보가 반복적으로 등장하였다. 이러한 특성은 기존 문서 구조를 유지하면서 신규사업 정보를 반영하여 계획서 초안을 작성하는 가능성을 제시한다. 하지만 실제 작성 시 반복적인 변경 대상 표현의 탐색 및 수정 등 행위는 작성자의 검토 부담을 증가시킨다.

LLM(large language model)은 이러한 문서 개정에 활용될 수 있지만, 기존 문서의 정보를 임의로 변경하는 구조 훼손 가능성이 있다[3]. 이에 본 연구에서는 LLM의 역할을 문서 재작성에 한정하지 않고, 수정 후보 식별 및 제안 수단으로 제한하여 사용자가 승인한 표현만 치환규칙으로 반영하는 문서 개정 방식을 제안한다. 특히 장문 문서 개정 시 LLM 단독 문서 개정 방식과 LLM 및 사용자

승인 기반 치환규칙 방식을 비교 분석하였다. 이를 통해 사용자 승인과 치환규칙을 결합한 문서 개정 방식의 활용 가능성을 확인한다.

2. 문서 개정 도구 구현 환경

방산업체의 업무 특성상 문서 개정 도구는 내부 또는 폐쇄망 환경에서 구동되어야 한다. 이에 본 연구에서는 별도 외부 웹 서버를 구축하지 않고 실행 가능한 Python 기반 문서 개정 도구를 구현하였다. JupyterLab에서 Python이 실행되며, LLM API와 연동하여 구동되도록 구성하였다. 문서 구조 보존과 텍스트 중심으로 문서를 개정하기 위해 개정 대상 문서는 Markdown 형식으로 변환하여 사용하였다. LLM 서비스 연동은 사내에서 제공하는 LLM API를 호출하는 방식으로 해당 서버로 요청을 전송 및 수신한다. 보안성을 고려하여 입력 문서에 포함된 기존 사업명, 체계명, 기관명, 부대명 등 주요 식별정보를 가명화하여 적용하였다. 게다가 제안 도구는 입력 문서를 LLM 서버에 상시 탑재하거나 학습시키는 방식이 아닌 문서 개정 과정에서 필요한 청크와 개정 입력자료를 LLM API 요청에 활용하는 방식으로 구성하였다.

3. 제안 도구 설계 및 구현

3.1 목적

본 연구에서 제안하는 문서 개정 도구는 LLM 및 치환규칙 모델로 기존 수명주기관리계획서의 문서 구조를 유지하면서, 신규사업 정보 자료를 기반으로 개정이 필요한 표현을 식별하고 사용자가 승인한 항목만 실제 문서에 반영하는 것을 목표로 한다. 기존 문서를 LLM이 전체 재작성하는 방식은 문맥 기반 수정에는 유리할 수 있으나, 장문 문서 처리 과정에서 원문 구조 훼손, 불필요한 문장 재작성, 입력 자료에 없는 내용 생성, 변경 이력 추적의 어려움이 발생할 수 있다[4]. 따라서 LLM을 문서 전체 수정 주체가 아니라, 개정 후보를 탐색하고 제안하는 보조 수단으로 제한하였다.

제안 모델의 활용 절차는 기존 문서와 개정 입력 자료를 입력받은 후, LLM이 신규사업 정보와 관련된 개정 후보를 도출한다. 그리고 사용자가 후보별 반영

여부와 최종 수정값을 검토 및 승인한 뒤, Python 기반으로 안전치환 모듈이 승인된 기존표현-최종수정값 규칙만 기존 문서에 적용하는 방식이다.

3.2 요구사항

위 모델은 개정 과정에서 수정 대상과 보존 대상을 구분할 수 있어야 한다. 또한 동일한 표현이라도 문맥에 따라 수정 여부가 달라질 수 있으며, 짧거나 범용적인 표현은 자동 치환 시 오수정으로 이어질 가능성이 있다. 따라서 제안 모델은 아래 Table 1과 같이 설계 요구사항을 도출하였다.

Table 1. Design requirements

Requirement	Description
Document Structure Preservation	Preserve the existing Markdown structure as much as possible
Modification Target Identification	Distinguish expressions to be revised from those to be preserved
LLM Role Restriction	Use the LLM only to identify revision input data and revision candidates
User Review and Approval	Apply only candidates reviewed and approved by the user
Replacement Rule Generation	Generate replacement rules from candidates approved by the user
Safe Execution	Apply approved rules to the Markdown document using Python string processing
Incorrect Replacement Prevention	Exclude ambiguous expressions with a high risk of incorrect replacement
Change History Logging	Record changes from original to final revised expressions
Review Report Generation	Report applied and excluded rules, changes, remaining candidates and normalization status

3.3 기능 구성

본 모델은 문서 처리 모듈, 개정 입력 자료 인식 모듈, 후보 탐지 모듈, 사용자 승인 모듈, 안전치환 모듈, 검증·리포트 모듈로 구성하였다. 이에 따른 LLM 및 치환규칙 모델의 흐름도는 Fig. 1과 같다.

3.3.1 문서 처리 모듈

문서 처리 모듈은 Markdown 형식의 기존 문서와

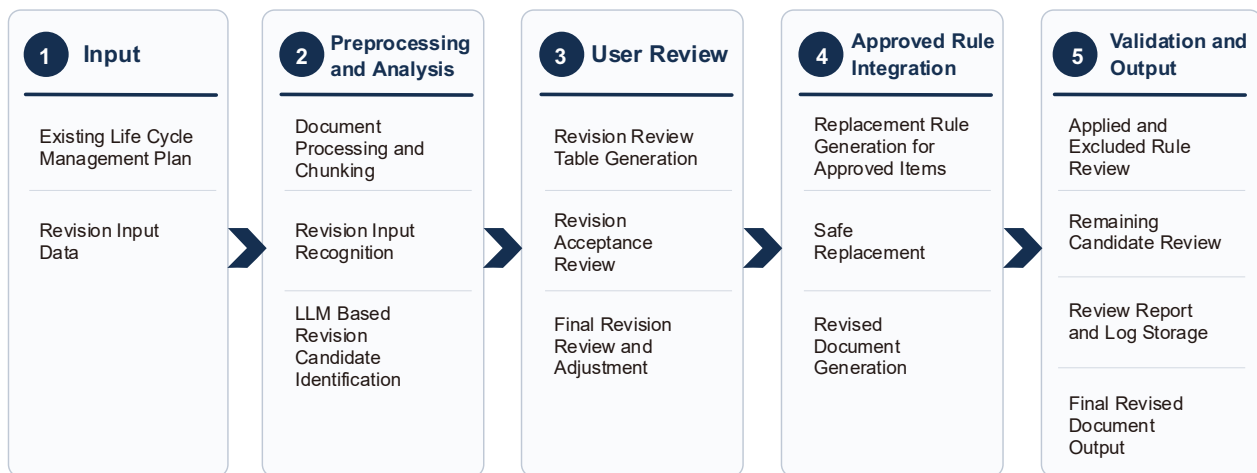


Fig. 1. Workflow of the LLM and replacement rule model

개정 입력 자료를 인식하여 LLM 및 Python 처리에 적합한 형태로 변환한다. 장문 문서를 LLM에 한 번에 입력하기 어렵기 때문에 기존 문서는 일정 길이의 청크로 분할된다. 청크 분할 시 청크 경계에서 문맥이 단절되는 문제를 줄이기 위해 문단, 표, 제목 등의 의미 단위가 과도하게 훼손되지 않도록 오버랩을 설정했다.

Table 2. Revision input items

Category	Recognition Targets
Project Identification Information	Project name, system name, equipment name, target system, customer organization, operating organization, and implementing organization
Project Schedule Information	Project period, contract period, development period, delivery schedule, test and evaluation, training, and technical manual preparation schedule
Project Scope	Project objectives, development scope, delivery scope, performance improvement scope, and systems to be interfaced
ILS/IPS Information	Scope of maintenance support, supply support, training, technical support, technical manuals, repair parts, support equipment, configuration management, obsolescence management, and system diagnostics
Deliverable Information	Information provided in plans, manuals, and other deliverables

3.3.2 개정 입력 자료 인식 모듈

개정 입력 자료 인식 모듈은 사용자가 입력한 신규 사업 정보 자료에서 문서 개정에 활용할 핵심 항목을 구조화한다. 개정 입력 자료는 별도의 규정 문서가 아닌 신규사업의 사업 식별 정보, 일정 정보, 사업 범위, ILS/IPS 정보[5], 산출물 정보를 포함하는 단일 자료로 Table 2와 같이 구성했다[6]. 개정 입력 자료에 기존 문서에서 찾을 표현 예시가 포함되어 있지 않은 경우, 신규사업 정보만을 기준으로 기존 문서 내 대응 표현을 탐색한다.

3.3.3 후보 탐지 모듈

후보 탐지 모듈은 구조화된 개정 입력 자료를 바탕으로 기존 문서에서 신규사업 정보와 대응될 가능성이 있는 기존 표현을 탐색한다. 기존 표현을 수정 후보로 식별하는 과정이며 실제 수정 여부는 이후 사용자 승인 모듈에서 결정한다.

3.3.4 사용자 승인 모듈

사용자 승인 모듈은 후보 탐지 모듈에서 도출된 수정 후보를 사용자가 검토하고 승인하는 단계이다[7]. 본 모델에서는 JupyterLab UI 내 카드형 검토표 형태로 후보를 제공한다. 사용자는 각 후보에 대해 반영 여부를 결정하고 필요 시 기존표현 또는 최종 수정값을 직접 수정할 수 있다. 이 단계에서 승인된 후보만

실제 치환 규칙으로 변환되며, 승인되지 않은 후보는 최종 문서에 반영되지 않는다.

3.3.5 안전치환 모듈

안전치환 모듈은 사용자 승인 모듈에서 확정된 수정 항목을 기존 **Markdown** 문서에 반영한다. LLM은 수정 후보를 식별하고 제안하는 역할을 담당하지만, 문서 변경은 **Python** 기반 문자열 처리 로직을 통해 수행된다. 치환 시 기존 표현의 존재 여부, 일반적인 단어 선택, 단문 표현의 오치환, 문서 구조 훼손 가능성 등을 고려하여 부분 치환이나 반복 치환으로 인한 오류를 최대한 방지한다.

3.3.6 검증·리포트 모듈

검증·리포트 모듈은 문서 개정 결과에 대한 검증 단계로 안전치환 모듈을 통해 생성된 수정본을 대상으로 승인 규칙의 적용 횟수, 청크별 처리 상태, 검증 실패 여부, 잔존 표현 후보 등의 현황을 **Json** 로그와 **Markdown** 검토 리포트 형식으로 저장한다. 이를 통해 수정 과정과 결과를 사용자가 재검토할 수 있는 형태로 제공한다. 검토 리포트에는 총 청크 수, 실제 적용 수, 정확 치환 적용 수, LLM 승인 적용 수, 제외된 치환 규칙, 사전·최종 잔존 후보 수 등이 포함된다.

4. 문서 개정 결과 분석

본 연구에서는 잠수함 관련 사업의 수명주기관리 계획서와 개정 입력 자료를 동일하게 적용하여 LLM 단독모델과 본 연구에서 제안하는 LLM 및 치환규칙 모델의 개정 결과를 확인하였다. LLM 단독모델은 기존 문서와 개정 입력 자료를 입력받은 뒤 LLM이 문서 청크를 직접 재작성하는 방식이다. 별도의 사용자 승인 절차나 치환 규칙을 사용하지 않으며, LLM이 생성한 결과를 수정본에 반영한다. LLM 및 치환규칙 모델은 LLM이 기존 문서에서 개정 후보를 탐색하고 수정 후보를 제안한 뒤, 사용자가 승인한 항목만 실제 문서에 반영하는 방식이다. 이 모델에서는 사용자가 승인한 기존표현과 변경 후 표현의 쌍을 치환 규칙으로 사용하며, 안전치환 모듈이 치환을 수행한다.

Table 3. Comparison of model configurations

Category	Standalone LLM Model	LLM and Replacement Rule Model
Input Document	Existing document and revision input data	Existing document and revision input data
LLM Role	Directly revises document content	Identifies revision candidates and suggests revisions
User Approval	Not required	Required before modification
Replacement Rule	Not used	Generated from approved candidates
Actual Modification Method	Applies LLM output directly	Applies approved rules using Python string processing

4.1 LLM 단독모델

본 모델은 총 28개 청크를 처리하였으며, 이 중 정상 처리된 청크는 11개이고 17개 청크는 비정상 출력으로 감지되어 원문 보존 처리되었다. 전체 청크 중 유효 출력 처리율은 39.3%, 비정상 출력 감지율 및 원문 보존 처리율은 60.7%로 나타났다.

Table 4. Processing results of the standalone LLM model

Result Category	Value	Description
Total Chunk Count	28	Processing units generated by splitting the existing document
Number of Valid Output Chunks	11	Chunks whose LLM output was applied to the final revision
Number of Invalid Output Chunks	17	Chunks classified as invalid due to repeated tokens, repeated symbols, or empty output
Number of Preserved Source Chunks	17	Chunks for which the source text was preserved due to invalid output
Change in Character Count	-14,518	Decrease in character count compared with the input

4.2 LLM 및 치환규칙 모델

본 모델은 총 21개 청크를 처리하였다. 실제 적용된 변경은 총 50건으로 나타났으며, 이 중 정확 치환 적용 항목은 20건, LLM 승인 적용 항목은 30건이었다. 또한 수정 전 사전 잔존 후보는 203건이었고, 수정 후 최종 잔존 후보는 148건으로 나타났다.

Table 5. Processing results of the LLM and replacement rule model

Result Category	Value	Description
Total Chunk Count	21	Processing units generated by splitting the existing document
Number of Applied Changes	50	Changes applied to the document based on approved rules
Number of Exact Replacements	20	Replacements applied according to predefined rules
Number of LLM Replacements Approved by the User	30	LLM suggested candidates approved by the user and applied to the document
Initial Remaining Candidate Count	203	Candidates that may contain existing expressions before modification
Final Remaining Candidate Count	148	Remaining candidates requiring further review after modification

4.3 비교 분석

두 모델은 문서 개정 방식과 결과 해석 기준이 서로 다르기 때문에 단순 수치 비교보다 출력 안정성, 변경 추적성, 사용자 통제성, 후속 검토 가능성 측면에서 해석할 필요가 있다. 입력 자료 중 사업 식별 정보 등에 해당하는 짧은 단어 및 문구는 두 모델에서 대부분 개정 대상으로 반영되었다. 다만 LLM 단독모델은 일부 청크에서 비정상 출력이 발생하여 적용 결과의 안정성이 제한되었고, LLM 및 치환규칙 모델은 승인된 표현 단위의 변경이 안정적으로 적용되었다.

ILS/IPS 및 산출물 정보처럼 길고 복잡한 문장 또는 문단 단위의 정보는 장문 정보의 반영 측면에서는 두 모델의 차이가 명확하게 나타났다. LLM 단독모델은 장문 정보를 직접 재작성할 수 있으나 출력 일관성과 구조 보존 측면에서 불안정성이 확인되었다. 반면 LLM 및 치환규칙 모델은 사용자 승인 기반으로 안정적인 치환을 수행할 수 있으나 문장 자체를 재구성하거나 장문 정보가 반영되는 것은 제한되었다.

전반적으로 문서 본문을 LLM이 직접 생성하는 LLM 단독모델은 출력 품질이 모델 응답에 크게 의존한다. 일부 청크에서 동일 토큰의 반복, 과도한 기호 반복, 비정상 문자열 생성 등이 발생하였다. 이는 LLM이 장문 문서를 청크 단위로 직접 재작성하는 과정에서 입력 문맥을 안정적으로 유지하지 못하거나,

Table 6. Comparison of model processing results

Category	Standalone LLM Model	LLM and Replacement Rule Model
Output Stability	Depends on output generated by the LLM	Reduces output variability based on approved rules
Change Traceability	Limited tracking of individual changes	Enables tracking through applied rules and change history
User Control	No prior approval process	User approval required before modification
Subsequent Review Capability	Comparison mainly based on generated output	Enables review of applied rules, excluded rules, and remaining candidates
Long Document Processing	Possible but consistency is reduced	Limited by context length and document reconstruction

출력 생성이 특정 토큰 패턴으로 수렴한 결과로 볼 수 있다. LLM 및 치환규칙 모델은 어떤 표현이 어떤 값으로 변경되었는지, 해당 표현이 몇 회 적용되었는지, 어떤 규칙이 제외되었는지를 검토 리포트에서 확인할 수 있다. 이 점에서 LLM 단독모델에 비해 변경 추적성과 검토 가능성을 확보하기 용이하다. 또한 사용자가 변경 범위를 통제하고 승인된 규칙만 문서에 반영하기 때문에 비정상 출력이나 불필요한 원문 재작성, 문서 구조 훼손 위험을 줄일 수 있다. 한편 최종 잔존 후보는 사전 잔존 후보 대비하여 크게 줄어들지 않았는데 수정 후에도 기존 표현이 남아 있을 가능성이 있어 사용자가 후속 검토해야 하는 후보로 해석하는 것이 적절하다.

5. 고찰

본 연구의 결과는 수명주기관리계획서와 같은 장문 기술문서 개정에서 LLM을 어떤 역할로 활용할 것 인지가 중요하다는 점을 보여준다. LLM을 문서 본문을 직접 생성하는 주체로 사용할 경우 문맥을 반영한 자연스러운 개정이 가능할 수 있으나, 출력 결과가 모델 응답에 크게 의존하게 된다[8]. 특히 장문 문서를 청크 단위로 처리하는 과정에서는 반복 출력, 빈 출력, 과도한 기호 반복 등 비정상 출력이 발생할 수 있으며, 이는 문서 구조 보존과 개정 결과의 신뢰성 측면에서 부담으로 작용할 수 있다.

LLM 및 치환규칙 방식은 LLM의 역할을 후보 탐지와 수정 제안으로 제한하고, 실제 문서 변경은 사용자 승인과 규칙 기반 안전 치환 절차를 통해 수행하도록 구성하였다[9]. 이는 LLM의 문맥 해석 능력을 활용하면서도, 최종 반영 단계에서는 사용자의 판단과 승인된 규칙을 통해 결정성과 통제성을 확보하기 위한 방식이다[10]. 또한 어떤 표현이 어떤 값으로 변경되었는지, 몇 회 적용되었는지, 어떤 규칙이 제외되었는지를 검토 리포트에서 확인할 수 있다는 점에서 변경 추적성과 후속 검토 가능성을 제공한다. 따라서 문서 전체를 자동으로 완성하는 방식이라기보다 선별적 개정을 지원하는 방식으로 해석할 필요가 있다. 비교적 짧고 명확한 표현은 치환규칙으로 반영하기 용이하지만, ILS/IPS 세부 수행 범위나 산출물 정보처럼 문장 또는 문단 수준의 해석과 재구성이 필요한 항목은 단순 치환만으로 충분히 반영되기 어렵다. 이러한 점에서 제안 방식의 의의는 자동으로 완성된 정답 문서를 생성하는 데 있는 것이 아닌 수정 후보와 적용 이력, 제외 규칙, 잔존 후보를 구조화하여 사용자가 검토 가능한 형태로 제공하는 것에 있다.

6. 결론

본 논문에서는 수명주기관리계획서의 개정을 지원하기 위해 LLM 기반 후보 탐색과 사용자 승인형 치환규칙을 결합한 문서 처리 방안을 제안하고, 이를 도구 형태로 구현하였다. 제안 방식은 기존 문서와 개정 입력자료를 입력받아 LLM이 개정 후보를 탐색하고, 사용자가 후보별 반영 여부와 최종 수정값을 검토·승인한 뒤, 승인된 기존표현과 최종 수정값의 쌍을 치환규칙으로 변환하여 문서에 반영하는 절차로 구성된다. 동일한 기존 문서와 개정 입력자료를 대상으로 LLM 단독 모델과 LLM 및 치환규칙 모델을 적용한 결과, 두 방식은 서로 다른 특성을 보였다. LLM 단독 모델은 문서 청크를 직접 재작성할 수 있으나, 일부 청크에서 비정상 출력이 발생하였고 개별 변경 사항을 규칙 단위로 추적하기 어려웠다. 반면 LLM 및 치환규칙 모델은 승인된 표현 단위의 변경을 적용하고, 적용 규칙, 제외 규칙, 잔존 후보를 검토 리포트로 제시함으로써 사용자가 수정 결과를 확인하고 후속 검토할 수 있는 구조를 제공하였다.

수명주기관리계획서와 같은 장문 기술문서에서는

LLM을 완전 자동 작성 도구로 활용하기보다 사용자의 판단을 보조하는 후보 탐색 도구로 활용하는 방식이 선별적 개정 업무에 실무적으로 활용될 수 있음을 확인하였다. 특히 기존 문서 구조와 검토 이력이 중요한 문서 개정에서 사용자가 변경 범위를 통제하고 승인된 규칙만 반영하는 절차가 변경 추적성과 검토 가능성이 확보하는 데 도움이 된다.

향후에는 다양한 사업에 제안 방식을 적용하여 활용 가능성을 추가로 검토할 필요가 있다. 현재 방식은 짧고 명확한 표현의 치환에 중점을 두고 있으므로, 문장 또는 문단 단위의 보완이 필요한 항목을 반영하기 위해 절 단위 보완 및 사용자 승인 기반 문장 재작성 기능을 구현할 필요가 있다. 아울러 치환 시 위험도 표시, 유사 및 중복 표현 경고, 문서 구조 보존성 평가 기능을 보완한다면 제안 도구의 실무 적용성을 더욱 높일 수 있을 것으로 기대된다.

참고문헌

- [1] Ministry of National Defense, Directive on Total Life Cycle Management, MND Directive No. 3144, National Law Information Center, 2026.
- [2] Defense Acquisition Program Administration, Regulation on Life Cycle Management in the Acquisition Phase, DAPA Directive No. 960, National Law Information Center, 2026.
- [3] Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, & Pascale Fung, 'Survey of Hallucination in Natural Language Generation,' ACM Computing Surveys, VOL. 55, NO. 12, 2023, pp. 1-38.
- [4] Joshua Maynez, Shashi Narayan, Bernd Bohnet, & Ryan McDonald, 'On Faithfulness and Factuality in Abstractive Summarization,' in Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, July 2020, pp. 1906-1919.
- [5] Defense Acquisition University, Integrated Product Support (IPS) Elements Guidebook, Version 1.0b, 2024.
- [6] Defense Acquisition Program Administration, Standard Special Contract Conditions for Naval Shipbuilding (General and Defense Industry), DAPA Established Rule No. 1075, National Law Information Center, 2026.
- [7] Eduardo Mosqueira-Rey, Elena Hernández-Pereira, David Alonso-Ríos, José Bobes-Bascarán, & Ángel Fernández-Leal, 'Human-in-the-Loop Machine Learning: A State of the Art,' Artificial Intelligence Review, VOL. 56, 2023, pp. 3005-3054.
- [8] Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin et al., 'A Survey on

Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions,' ACM Transactions on Information Systems, VOL. 43, NO. 2, 2025, Article 42.

[9] Konstantinos Lazaros, Aristidis G. Vrahatis, & Sotiris Kotsiantis, 'Human-in-the-Loop Artificial Intelligence: A

Systematic Review of Concepts, Methods, and Applications,' Entropy, VOL. 28, NO. 4, 2026, Article 377.

[10] Ben Shneiderman, 'Human-Centered Artificial Intelligence: Reliable, Safe & Trustworthy,' International Journal of Human-Computer Interaction, VOL. 36, NO. 6, 2020, pp. 495-504.