Original Article

https://doi.org/10.12985/ksaa.2026.34.2.011
ISSN 1225-9705(print) ISSN 2466-1791(online)

항공안전관리시스템(SMS) 자율보고서의 Hazard 표준코드 자동 분류를 위한 LLM·RAG 기반 모델의 정확도 검증 연구

임세훈*

A Study on the Accuracy Validation of LLM·RAG-Based Models for Automated Classification of Standard Hazard Codes in Aviation SMS Voluntary Reports

Se Hoon Yim*

ABSTRACT

This study examined the accuracy of LLM and RAG based models for automatically classifying hazard standard codes from unstructured Safety Management System, SMS, voluntary safety reports. A retrospective comparative design was applied to 50 reports collected from a university aviation training organization, using expert assigned hazard codes as ground truth. ChatGPT 5.2 thinking, Gemini 3.1 Pro, NotebookLM, and Google AI Studio based on Gemini 3 Flash were evaluated in a multilabel classification setting with accuracy, precision, recall, and F1-score. NotebookLM showed the most balanced performance, with 80.0% accuracy, 70.0% precision, and 56.8% F1-score, while Gemini 3.1 Pro showed the highest recall at 48.9%. These findings suggest that LLM and RAG approaches can support initial hazard coding in SMS practice, although expert review and refinement of the current hazard taxonomy are still required for reliable operational use.

Key Words : Safety Management System(항공안전관리시스템), Hazard Code Classification(위해요인 표준코드 분류), Large Language Model(대형언어모델), Retrieval-Augmented Generation(검색증강생성), Voluntary Safety Report(자율안전보고서)

Ⅰ. 서 론

국제민간항공기구(ICAO)는 Annex 19에 따라 회원국 및 관련 항공 서비스 제공자들에게 안전관리시스템(safety management system, SMS)의 운영을 의무화하고 있다. 국내의 경우에도 항공안전법 제58조 2항 각호에 따라 전문교육기관 등은 SMS를 마련하고 국토교통부장관의 승인을 받아 운용하도록 하고 있으며 항공안전데이터를 체계적으로 관리하여야 한다.

전문교육기관은 항공안전법 제58조 2항 2호에 따라 항공종사자의 의무보고 및 자율보고 등을 통해 항공안전데이터 정보가 수집되고 이때 수집된 항공안전데이터는 표준 분류체계로 데이터 일관성을 확보해야 한다. 항공안전데이터는 표준코드로 변환되어야만 특정 기상

Received: 24. Mar. 2026, Revised: 6. May. 2026, Accepted: 26. May. 2026
* 한서대학교 헬리콥터조종학과 부교수
연락저자 E-mail : SHProf@hanseo.ac.kr
연락저자 주소 : 충남 태안군 남면 곰섬로 236-49, 한서대학교 태안비행장 본관 227호

조건 또는 특정 기종에서 반복적으로 발생하는 위험의 패턴을 정확히 분석하고 선제적인 예방 조치를 취할 수 있기 때문이다.

항공기 사고, 준사고는 CICTT(CAST/ICAO common taxonomy team) 분류법을 활용해 발생 유형(occurrence category), 비행 단계(event phase) 등으로 분류하고 위해요인(hazard)은 항공안전데이터 처리 및 활용에 관한 규정(행정규칙) 별표 8 및 별표 8의 2에 따라 항공안전데이터를 표준분류하고 있다.

그런데 공항이 아닌 비행장이나 민간 전문교육기관과 같이 소규모 조직의 경우 이러한 데이터를 수작업으로 코딩할 수 있는 안전관리 전담인력이 부족하여 수집된 보고서를 일일이 표준 분류코드로 정리하기 어렵다. 게다가 자율보고서는 자유서술형 텍스트로 작성되며 보고자마다 표현 방식이 달라 일관된 데이터 분석이 어렵다는 한계가 있다.

최근 인공지능 분야에서 대형언어모델(large language model, LLM)이 비약적인 발전을 거듭하고 2026년에는 고도화된 추론 역량을 확보함에 따라 서술식 텍스트의 맥락을 이해하고 이를 표준코드로 자동 매핑하는 기술적 대안이 대두되고 있다.

이에 본 연구는 인력과 전문성이 제한된 소규모 항공 조직이나 민간 전문교육기관 환경을 고려하여 비정형 자율보고서의 위해요인(hazard) 표준코드 분류를 자동화하기 위한 대안으로서 환각(hallucination) 완화 기법으로 알려진 검색증강생성(retrieval-augmented generation, RAG) 기반 LLM의 적용 가능성과 정확도를 검증하는 것을 목적으로 한다.

II. 이론적 배경

2.1 ICAO ADREP 및 CICTT 분류체계

국제민간항공기구(ICAO)의 ADREP(accident/ incident data reporting programme) 분류체계는 사고 및 준사고 유형을 표준화된 용어 및 코드로 표준분류를 제시하고 있는데 항공기 카테고리, 발생 유형(occurrence category), 사건 단계(event phase), 설명요인(explanatory factors) 등 다수의 속성과 값으로 구성된다(ICAO, 2026).

아울러 CICTT는 SMS에서 활용할 위해요인 분류를 위해 환경(environmental), 기술(technical), 조직(organizational), 인적(human)의 4가지 카테고리를 설정했다. 현재 미국 FAA, 유럽 EASA 뿐만 아니라 국내에서도 CICTT 분류체계를 표준으로 사용하고 있다.

표준 분류체계를 사용하면 서로 다른 보고서에서 동일한 사건을 동일한 코드로 식별할 수 있어 안전 추세 분석과 위험평가가 가능하다. 그러나 분류체계가 도입되어 있더라도 보고 주체와 시스템이 분산되어 있으면 데이터의 일관된 통합 관리는 별개의 과제로 남는다. Kim et al. (2020)은 이러한 문제를 지적하며 국내 항공안전데이터의 통합 관리를 위한 표준 데이터 구조의 정립이 필요하다고 제안하였다.

항공안전데이터의 분류 작업은 단순한 키워드 매칭 수준을 넘어선다. 동일한 사건이라 하더라도 사고 원인과 상황에 따라 복합 코드가 부여될 수 있으며 난기류와 같은 환경요인 또한 발생 원인에 따라 서로 다른 코드로 구분된다. 이와 같이 ICAO 분류체계는 구체적인 사용 지침에 따라 세밀한 코드 할당을 요구하므로 정확한 코딩을 위해서는 전문적인 이해와 해석이 필요하다.

대형 항공사의 경우 전문가 및 SMS 데이터 분석팀이 항공안전데이터를 분석하고 결과에 대한 교차 검증을 수행한다. 나사(NASA)의 보고서에 따르면 전문가가 이러한 보고서를 수작업으로 검토하는 데에는 많은 시간이 소요되며 보고서 1건당 최대 5일이 걸리기도 한다고 하였다(NASA, 2023).

더욱이 항공안전보고서는 비정형 구조의 자유 서술식 텍스트로 작성되는 경우가 많아 대규모 데이터 분석이 어려우며 보고서에 내재된 복잡한 맥락 정보를 충분히 반영하지 못하는 한계가 있다. 이러한 특성은 위험요소의 신속한 식별과 효과적인 안전 개선조치 도출을 저해하는 요인으로 작용할 수 있다(Zhou et al., 2024).

소규모 항공 조직이나 민간 전문교육기관에서는 제한된 인력과 조직 구조로 인해 복잡한 위해요인 분류 기준을 일관되게 적용하기 더욱 어려운 환경이다. 또한 보고자나 분류 담당자의 주관적 해석에 따라 동일 사건에 서로 다른 코드가 부여되는 분류 비일관성이 발생할 수 있으며 이는 항공안전데이터베이스의 데이터 품질과 무결성을 저하시키는 원인이 된다.

2.2 국내 자율보고 데이터 분류 체계

국내에서는 항공안전법 및 관련 시행규칙에서 의무보고 및 자율보고제도를 규정하며 나르미(NARMI)와 항공자율보고시스템(KAIRS) 등을 통해 데이터가 수집

되고 있다. 항공자율보고시스템(KAIRS)은 항공안전법 제61조에 근거하여 잠재적 위해요인을 자율적으로 보고하도록 하는 제도이다. 수집된 보고서는 운영기관에 의해 전문적으로 분석되며 보고자의 익명성이 보장된 상태로 항공종사자와 관련 기관에 공유되어 안전 개선 조치 수립과 항공사고 예방을 위한 기초 자료로 활용된다. 나르미는 국토교통부장관이 항공안전 활동의 과정 및 결과 등으로 생성된 안전데이터를 저장 및 기록하는 전자시스템이다(Park et al., 2022).

그러나 이러한 시스템에도 보고서 내용은 서술식으로 저장되어 후속 분석을 위해서는 별도의 분류 작업이 필요하다.

국내 항공안전 자율보고 데이터의 분류는 항공안전데이터 처리 및 활용에 관한 규정(행정규칙) 별표 8의 SM-ICG 분류체계와 별표 8의2에 제시된 한국형 위해요인 분류체계를 기반으로 수행된다. 별표 8의 SM-ICG 분류체계는 위해요인의 성격을 기준으로 조직적(organizational), 환경적(environmental), 인적(human), 기술적(technical) 요인의 네 가지 상위 범주로 구성되며 하위 단계에서는 운영영역과 유형에 따라 세부 분류가 이루어진다(MOLIT, 2024).

그러나 이 분류체계는 구조적 측면에서 몇 가지 한계를 가진다. Table 1과 같이 별표 8에 제시된 SM-ICG 분류체계는 일부 항목에서는 '운영' 항목에 감독당국, 항공사 운항, 공항 등 특정 조직이나 운영 영역이 포함되는 반면, 환경적 또는 인적요인의 경우 '전 분야'로 포괄적으로 정의되어 계층 간 일관성이 부족하다.

이로 인해 '운영' 항목이 항상 '유형'의 상위 범주로 작동하지 못하며 결과적으로 완전한 계층적 트리 구조를 형성하기 어렵다는 문제가 존재한다.

한편 별표 8의2에 제시된 한국형 위해요인 분류체계는 식별번호, 분야, 유형, 위해요인 명칭 및 설명 등으로 구성된 데이터베이스 형태의 구조를 가진다. 이 체계에서는 '분야'(운항, 공항, 관제 등)와 '유형'(human, OPS, ORG, weather 등)이 상하 계층 관계가 아니라 독립적인 속성으로 존재하는 매트릭스 구조를 형성하고 있어 분류 기준의 일관성을 확보하기 어렵다는 문제가 있다.

이러한 구조적 특성은 자율보고 데이터의 위해요인(hazard) 코딩 과정에서 분류 기준의 해석과 코드 선택에 대한 전문적 판단을 요구하며 특히 인력과 전문성이 제한된 소규모 항공 조직이나 민간 전문교육기관에서는 일관된 데이터 분류를 수행하는 데 어려움을 초래할 수 있다.

이러한 구조적 한계에도 불구하고 한국형 위해요인 분류체계는 현재 국내에 공식적으로 적용되는 유일한 분류기준이므로 실제 SMS 자율보고 데이터의 분석과 관리 과정에서는 이를 기반으로 한 코딩 작업이 수행될 수밖에 없다. 따라서 본 연구에서는 한국형 분류체계를 기준으로 정의된 270여 개의 위해요인 코드를 활용하여 데이터 분류 체계를 구성하고 이를 분석에 적용하였다. 다만 본 연구의 분석 과정에서 위해요인 코드 간 중복 및 정의 모호성 등 분류체계 자체의 추가적인 문제점도 함께 도출되었으며 이는 V장 논의에서 구체적으로 다룬다.

2.3 자연어 처리 기술의 진화

항공안전보고서 텍스트의 분류 자동화는 오랫동안 학

Table 1. Comparison of hazard taxonomy structures between attached table 8(SM-ICG) and the Korean hazard taxonomy

Category	Attached table 8(SM-ICG)	Category	Attached table 8-2(Korean hazard taxonomy)
Top-level category	Organizational, environmental, human, technical	Identification number	(Not applicable)
Operation	Airline operations & maintenance, airport operations, air navigation service providers, etc.	Domain	Flight operations, airport, air traffic control, maintenance, ground handling, etc.
Type	Management, Documentation, Weather, Facilities, etc.	Type	Human, OPS, Weather, ORG, Tech
Classification list	Detailed classification list	Hazard description	Detailed description of the hazard

계와 산업계의 숙원이었다. 과거의 연구들은 미국의 ASRS(aviation safety reporting system) 데이터 등을 대상으로 텍스트 분류, 원인요인 식별, 네트워크 기반 분석 등 고전적인 자연어 처리 기법을 폭넓게 적용해 왔다.

하지만 초기 형태의 신경망 모델들은 텍스트 내에 존재하는 단어의 출현 빈도나 단순한 통계적 패턴에 의존했기 때문에 문맥 내 깊이 숨겨진 인적 요인이나 복잡한 상황인식의 오류 같은 추상적 맥락을 파악하는 데 명확한 한계를 보였다(Tanguy et al., 2016; Yang et al., 2022).

이후 트랜스포머 아키텍처 기반의 BERT(bidirectional encoder representations from transformers) 모델이 등장하면서 문맥을 양방향으로 이해할 수 있는 길이 열렸다(Devlin et al., 2019; Nanyonga et al., 2025).

최근 항공안전데이터를 활용한 연구에서는 대규모 데이터베이스를 기반으로 다양한 자연어 처리 및 인공지능 기법이 적용되고 있다. 미국의 ASRS 보고서를 대상으로 LLM을 활용하여 사건 보고서를 요약하고 사고분석을 지원하는 연구가 수행된 바 있다(Basil, 2025).

또한 최근에는 항공사고 보고서를 대상으로 HFACS (human factors analysis and classification system) 분류를 자동화하기 위해 강화학습 기반 방법을 적용한 연구도 보고되고 있다(Ahmadi et al., 2025).

그러나 이러한 연구들은 대규모 학습 데이터 구축과 모델 학습을 전제로 하는 경우가 많아 인력과 데이터가 제한된 소규모 항공 조직이나 민간 전문교육기관에서 직접 적용하기에는 현실적인 제약이 존재한다. 그런데 2024년을 기점으로 LLM은 폭발적으로 성장하여 2026년 현재 고도화된 LLM 생태계는 이러한 패러다임을 완전히 뒤집었다. 1,750억 개 이상의 파라미터를 지닌 초창기 GPT-3의 성공을 넘어 현재의 GPT-thinking 5.4, Gemini 3.1 Pro, Claude Opus 4.6 등과 같은 모델들은 프롬프트(prompt)만으로 텍스트를 분석하고 추론하며 기존의 파인튜닝 모델을 압도하는 성능을 보여주고 있다.

그럼에도 불구하고 생성형 AI의 본질적인 한계인 환각(hallucination) 현상이 존재한다. 이 문제는 신뢰도가 중요한 SMS 자율보고 표준분류에 치명적이므로 완화 대책이 필요하다. 여러 연구에서 검색증강생성(RAG)이 환각 완화 기법으로 주목된다. 검색증강생성(RAG)은 입력 문장과 관련된 외부 문서를 검색하여 모델 입력으로 포함시키므로 모델이 생성하는 답변이 검색된 문서와 일치하도록 유도한다. 검색증강생성(RAG) 방식은 모델의 사전학습 파라미터만 사용하는 전통적인 LLM과 달리, 외부 문서나 데이터베이스에서 관련 정보를 동적으로 검색한 뒤 이를 바탕으로 응답을 생성하므로 사실성을 크게 개선하고 환각(hallucination)을 줄일 수 있다(Tonmoy et al., 2024; Bernardi et al., 2025).

최근 OpenAI, Google 등은 API의 File Search 기능을 발표하여 사용자가 파일이나 문서를 업로드하면 자동으로 임베딩, 인덱싱, 검색 등을 수행하고 이 검색된 내용만을 기반으로 답변을 생성하도록 지원한다(OpenAI, 2024; Google, 2025).

Ⅲ. 연구방법

3.1 연구설계

본 연구는 LLM이 비정형 자율보고서 텍스트로부터 위해요인 표준코드를 얼마나 정확하게 자동 분류할 수 있는지를 평가하기 위한 후향적 비교연구(retrospective comparative study)로 설계되었다. 이를 위해 동일한 자율보고 사례에 대해 인간 전문가가 부여한 위해요인 코드를 기준 정답(ground truth)으로 설정하고 동일한 입력 텍스트를 기반으로 LLM이 생성한 위해요인 코드와 이를 비교·평가하는 방식으로 연구를 수행하였다.

LLM 기반 분류 과정에서 발생할 수 있는 환각(hallucination) 문제를 완화하고 위해요인 코드 분류의 정확도와 신뢰성을 향상시키기 위해 본 연구에서는 검색증강생성(RAG) 기법을 적용하였다. 이를 위해 Fig. 1과 같이 SMS의 한국형 위해요인 분류체계에 따른 표준코드 정의서를 기반으로 지식 데이터베이스를 먼저 구축하였다. 해당 지식 데이터베이스는 위해요인 표준코드 정의와 분류 기준을 포함하며 검색증강생성(RAG) 구조를 통해 모델의 입력 컨텍스트에 제공되어 위해요인 기술(hazard description)의 의미 해석과 코드 선택과정에서 참조 정보로 활용되도록 설계하였다.

본 연구에서는 검색증강생성(RAG) 기반 분류 접근을 적용한 LLM 모델로 ChatGPT 5.2 thinking model, Gemini 3.1 Pro, NotebookLM(Google), Google AI Studio(Gemini 3 Flash)의 4가지 모델을

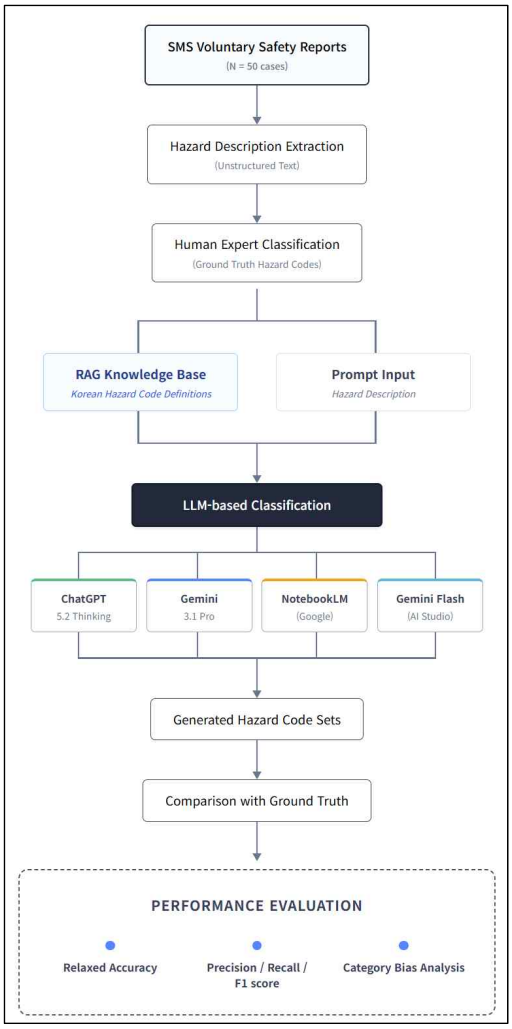

Fig. 1. Research design: framework for LLM-based hazard classification

비교 대상으로 선정하였다. 각 모델에는 동일한 프롬프트 구조를 적용하여 위해요인 표준코드를 생성하도록 하였으며 LLM 모델이 생성한 코드 결과는 인간 전문가의 분류 결과를 기준 정답(ground truth)으로 설정하여 분류성능을 평가하였다.

Table 2와 같이 프롬프트에는 다음과 같은 핵심 지침이 포함되었다. 첫째, 입력된 자율보고 데이터를 검토하여 위해요인의 명칭(hazard name), 내용(description), 대분류(category), 운영(operation), 유형(type), 목록(list) 등 6개의 항목을 추출하도록 지시하였다. 둘째, 각 항목의 분류 기준은 항공안전 자율보고 데이터 분류에 사용되는 별표 8 및 별표 8의2의 위해요인 분류체계를 기반으로 수행하도록 하였다. 셋째,

Table 2. Prompt template used in the study

Role :

You act as an aviation safety management expert responsible for constructing a hazard map based on SMS safety reports.

Objective :

Identify potential safety hazards and map them according to the official hazard taxonomy.

Tasks :

1. Extract the following fields:
   - Hazard Name
   - Description
   - Category
   - Operation
   - Type
   - List
2. Classify hazards according to the taxonomy defined in Attached Table 8 and Table 8-2.
3. Present the results in a structured table format.
4. Provide expert commentary on potential safety implications.
5. If information is missing, indicate "Insufficient data".

분석 결과는 표 형식으로 정리하여 구조화된 정보로 제공하도록 하였으며, 누락된 정보가 존재할 경우에는 데이터 미비로 표시하도록 지시하였다.

3.2 데이터 수집 및 전처리

본 연구에 사용된 데이터는 대학교 전문교육기관에서 수집된 최근 2년간의 SMS 자율보고서 서술형 데이터를 수집하였다. 최종적으로 분석에 사용된 데이터셋은 총 50건의 자율보고 사례로 구성되었다. 위해요인 표준코드는 항공안전 분야 전문가가 SMS 위해요인 분류체계에 따라 부여한 결과를 사용하였으며 본 연구에서는 해당 전문가 분류 결과를 기준 정답(ground truth)으로 간주하였다.

각 사례는 사례 식별번호(case identifier), 인간 전문가가 부여한 위해요인 코드 집합 그리고 각 LLM이 생성한 위해요인 코드 집합으로 구성된다. SMS 위해요인 분류체계에서는 하나의 사건이나 상황이 복수의 위해요인에 동시에 해당할 수 있으므로 단일 사례에 여러 개의 위해요인 코드가 부여될 수 있다. 따라서 본 연구의 분류 문제는 단일 클래스 분류가 아닌 다중라벨 분

류(multi-label classification) 문제로 정의하였다.

LLM 출력 결과와 인간 전문가의 분류 결과는 모두 문자열 형태로 저장되어 있었기 때문에 코드 단위 비교가 가능하도록 전처리 과정을 수행하였다. 수집된 위해요인 코드는 발생 요인의 속성에 따라 영문 접두사(prefix)와 세부 분류 숫자가 결합된 형태[예 : OP(운영), H(인적), T(기술), W(기상) 등]로 구성되어 있다. 그리고 각 셀에 포함된 문자열을 쉼표(,) 기준으로 분리하여 OP105, OP050, OR032와 같이 코드 목록 형태로 변환하였다. 이후 각 코드의 앞뒤 공백을 제거하고 동일 코드가 중복 포함된 경우 하나만 유지하도록 정리하였다. 또한 빈 문자열이나 분류체계에 존재하지 않는 비표준 코드 형식은 제거하였다.

본 연구에 사용된 자율보고 데이터는 연구 목적에 맞게 필요한 정보만을 활용하였으며 개인식별 정보가 포함된 경우에는 분석 과정에서 제외하였다.

3.3 분석 방법

모델의 분류 성능을 평가하기 위해 정확도(accuracy), 정밀도(precision), 재현율(recall) 그리고 F1-score를 산출하였다. 모델이 인간 전문가가 부여한 위해요인 코드 중 최소 하나 이상을 올바르게 포함하는지를 기준으로 정확도를 계산하고 정확도 추정의 불확실성을 고려하기 위해 95% Wilson 신뢰구간(confidence interval)을 함께 산출하였다. 그리고 다중라벨 분류 성능을 보다 정밀하게 평가하기 위해서 정밀도, 재현율 그리고 F1-score를 계산하였다.

$$Accuracy = \frac{correctly\ \ matched\ \ cases}{total\ \ cases} \quad (1)$$

$$Precision = \frac{correct\ predicted\ codes}{total\ predicted\ codes} \quad (2)$$

$$Recall = \frac{correct\ predicted\ codes}{total\ human\ codes} \quad (3)$$

$$F1 = \frac{2\mathrm{Pr}ecision \times Recall}{\mathrm{Pr}ecision + Recall} \quad (4)$$

정량적 성능 지표 외에도 모델이 특정한 위해요인 유형(hazard category)을 체계적으로 과대 또는 과소 예측하는지를 파악하기 위해 위해요인 코드의 접두사를 기준으로 유형별 예측 편향성을 분석하였다. SMS 위해요인 분류체계에서 위해요인 코드는 코드의 앞부분(Prefix)이 위해요인의 상위 범주를 나타내는 구조를 가지는데 예를 들어 OP는 operational, OR은 organizational, T는 technical, H는 human category를 의미한다. 본 연구에서는 이러한 코드 접두사를 기준으로 각 모델이 생성한 위해요인 코드의 유형 분포를 산출하고 이를 인간 전문가의 분류 결과와 비교하여 특정 유형에 대한 예측 편향성을 분석하였다.

Ⅳ. 연구결과

4.1 모델 간 성능평가 분석 결과

Table 3과 Fig. 2는 각 LLM 모델의 정확도와 성능을 비교한 결과이다. 정확도를 비교한 결과 NotebookLM(Google)이 80.0%로 가장 높은 값을 나타냈으며 이에 대한 95% Wilson 신뢰구간은 0.670에서 0.888 범위로 추정되었다. 이는 동일한 조건에서 반복적으로 표본을 추출할 경우 실제 정확도가 해당 구간 내에 포함될 가능성이 95% 수준임을 의미한다.

Gemini 3.1 Pro와 Google AI Studio(Gemini 3 Flash)는 각각 0.74의 정확도를 보였으며, 95%

Table 3. Accuracy comparison of LLM models with 95% Wilson confidence intervals

Model	Accuracy	95% Wilson CI
ChatGPT 5.2 thinking model	0.640(64.0%)	[0.501, 0.759]
Gemini 3.1 Pro	0.740(74.0%)	[0.604, 0.841]
NotebookLM Google	0.800(80.0%)	[0.670, 0.888]
Google AI studio(Gemini 3 Flash)	0.740(74.0%)	[0.604, 0.841]

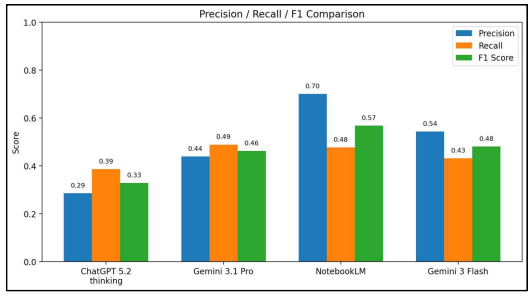

Fig. 2. Comparison of precision, recall, and F1-scores across LLM models

Wilson 신뢰구간은 모두 0.604~0.841로 나타났다. ChatGPT 5.2 thinking model은 0.64의 정확도를 보였으며, 95% Wilson 신뢰구간은 0.501~0.759 범위로 나타났다.

정밀도도 NotebookLM(Google)이 70.0%으로 가장 높은 값을 나타냈으며 Google AI Studio(Gemini 3 Flash)는 54.3%, Gemini 3.1 Pro는 43.9%, ChatGPT 5.2 thinking model은 28.6% 순으로 나타났다. 재현율은 Gemini 3.1 Pro가 48.9%로 가장 높은 값을 보였으며 NotebookLM(Google)은 47.7%, Google AI Studio(Gemini 3 Flash)는 43.2%, ChatGPT 5.2 thinking model은 38.6%으로 나타났다. F1-score는 NotebookLM(Google)이 56.8%로 가장 높은 값을 보였으며 Google AI Studio(Gemini 3 Flash)는 48.1%, Gemini 3.1 Pro는 46.2%, ChatGPT 5.2 thinking model은 32.9% 순으로 나타났다.

종합적으로 NotebookLM(Google)은 정밀도와 F1-score에서 가장 높은 값을 보여 가장 균형 잡힌 코드분류 성능을 나타냈으며 Gemini 3.1 Pro는 높은 재현율을 통해 전문가가 부여한 위해요인 코드를 가장 많이 회수하는 경향을 보였다.

반면 ChatGPT 5.2 thinking model은 가장 낮은 정밀도와 F1-score로 분석되었다.

4.2 코드 생성 특성

Table 4는 각 모델이 자율보고의 사례당 생성한 위해요인 코드 수의 분포와 과다 생성 비율을 비교한 결과를 나타낸다. 인간 전문가 분류 결과의 경우 사례당 평균 위해요인 코드 수는 1.76개였으며 중앙값은 2.0, 범위는 1~3개로 나타났다.

LLM 모델별 평균 생성 코드 수를 비교하면 ChatGPT 5.2 thinking model이 2.38개로 가장 높은 값을 보였으며 중앙값은 2.0, 범위는 1~4개로 나타났다. Gemini 3.1 Pro는 평균 1.96개, 중앙값 2.0, 범위 1~3개로 전문가 분포와 비교적 유사한 수준을 보였다. NotebookLM(Google)은 평균 1.20개로 가장 낮은 값을 나타냈으며 중앙값은 1.0, 범위는 1~3개였다. Google AI Studio(Gemini 3 Flash)는 평균 1.4개, 중앙값 1.0, 범위 1~3개로 나타났다.

코드의 과다 생성비율을 보면 ChatGPT 5.2 thinking model이 40.0%로 가장 높은 값을 보였으며 Gemini 3.1 Pro 역시 38.0%로 높은 수준을 나타냈다.

Table 4. Code generation characteristics

Model	Mean number of generated codes	Median	Range (min~max)	Over-generation rate
Human (ground truth)	1.76	2.0	1~3	-
ChatGPT 5.2	2.38	2.0	1~4	40.0%
Gemini 3.1 Pro	1.96	2.0	1~3	38.0%
NotebookLM Google	1.20	1.0	1~3	12.0%
Google AI Studio (Gemini 3 Flash)	1.40	1.0	1~3	16.0%

이는 두 모델이 인간 전문가보다 더 많은 위해요인 코드를 생성하는 경향이 비교적 빈번하게 나타남을 의미한다.

반면 NotebookLM(Google)은 12.0%, Google AI Studio(Gemini 3 Flash)는 16.0%로 상대적으로 낮은 과다 생성 비율을 보였다. 앞의 모델 성능평가에서 가장 높은 정밀도(precision)와 F1-score를 보였던 NotebookLM(Google)은 평균 생성 코드 수 역시 1.20개로 가장 낮게 나타났다. 이는 NotebookLM(Google)이 불확실한 코드를 추가로 생성하기보다는 비교적 확실성이 높은 코드만 선택하는 보수적인 전략을 취하는 것으로 해석된다.

4.3 카테고리 예측 편향성 분석

본 연구에서는 위해요인 코드의 접두사가 위해요인의 상위 범주를 나타내는 구조를 이용하여 OP(operational), H(human), T(technical), OR(organizational) 등의 카테고리 분포를 기준으로 모델의 카테고리 예측 편향을 분석하였다.

그 결과 Table 5와 같이 인간 전문가가 분류한 코드는 OP(운영)이 전체의 73.9%로 가장 높은 비중을 차지하였으며 H(인적)이 19.3%로 그 다음을 차지하였다. 반면 T(기술), OT(기타), W(기상) 범주는 각각 2.3% 수준으로 매우 낮은 빈도를 보였다.

모델별 카테고리 분포를 비교한 결과 Gemini 3.1 Pro가 전체적으로 인간 기준 분포와 가장 유사한 경향

Table 5. Comparative distribution of hazard categories between human experts and LLM models

Category	Human	ChatGPT 5.2	Gemini 3.1 Pro	NotebookLM Google	Google AI Studio (Gemini 3 Flash)
OP(operational)	73.9%	73.9%	69.4%	68.3%	47.1%
H(human)	19.3%	15.1%	20.4%	23.3%	34.3%
W(weather)	2.3%	0.8%	2.0%	3.3%	2.9%
T(technical)	2.3%	0.0%	6.1%	3.3%	10.0%
OT(other)	2.3%	0.0%	2.0%	0.0%	0.0%
OR(organizational)	0.0%	10.1%	0.0%	1.7%	5.7%

을 보였다.

다만 T(기술) 카테고리 분류는 높은 비율이 나타나 기술적 위해요인에 대한 예측이 다소 증가하는 경향이 확인되었다.

NotebookLM(Google)은 H(인적) 범주가 다소 과대 표현되는 비율이 나타났으나 전반적으로 인간 전문가와 비교적 유사한 패턴을 유지하였다.

ChatGPT 5.2 thinking model의 경우 OP(운영)의 비율(73.9%)은 인간 전문가 기준과 동일하게 나타났으나 인간 전문가가 분류하지 않은 OR(조직) 분류 코드를 10.1% 생성하는 것으로 확인되었다. 이는 실제 데이터에 존재하지 않는 카테고리를 생성한 것인데 카테고리 범주를 넓게 포함하는 경향이 있다고 해석되며 이러한 현상은 위의 분석에서 확인된 많은 코드 생성 수와도 일관된 결과를 보인다.

가장 큰 카테고리 예측 편향은 Google AI Studio (Gemini 3 Flash)에서 나타났다. 해당 모델은 OP(운영) 카테고리를 매우 낮은 비율로 예측하는 반면 H(인적)와 T(기술) 카테고리의 비중을 상대적으로 높게 생성하는 비율을 보였다.

종합적으로 볼 때 Gemini 3.1 Pro와 NotebookLM(Google)은 인간 전문가와 비교적 유사한 카테고리 예측 분포를 보인 반면 ChatGPT 5.2 thinking model, Google AI Studio(Gemini 3 Flash)는 강한 편향을 보이는 것으로 분석된다.

V. 논 의

본 연구는 검색증강생성(RAG) 기반 LLM을 활용하여 비정형 항공안전 자율보고서에서 위해요인 표준코드를 자동으로 분류하는 모델의 성능을 검증하였다. 성능평가 및 카테고리 편향성 분석 결과를 바탕으로 도출된 주요 시사점은 다음과 같다.

첫째, LLM의 코드 과다생성 현상은 단순한 오류가 아닌 새로운 위험 식별의 기회로 해석될 수 있다. 분석 결과 ChatGPT 5.2 thinking model과 Gemini 3.1 Pro 등 일부 모델은 인간 전문가보다 더 많은 수의 코드를 생성하는 경향을 보였다. 그러나 이러한 생성 결과물들을 심층 분석한 결과 단순히 환각(hallucination)에 의한 오분류라기보다는 인간 전문가가 수작업 검토 과정에서 미처 발견하지 못하거나 누락한 잠재적 위해요인을 LLM이 찾아낸 사례가 일부 존재하는 것으로 확인되었다. 이는 LLM이 인간의 인지적 사각지대를 보완하고 안전보고서를 교차 검증할 수 있는 유용한 도구로 활용될 수 있는 긍정적인 계기를 시사한다.

둘째, 현재 국내에서 사용 중인 한국형 위해요인 분류체계 자체의 구조적 모순과 중복성 개선이 시급하다. 연구 과정에서 분류 기준 정의서(별표 8의2) 내에 동일하거나 매우 유사한 위해요인이 각기 다른 코드로 산재되어 있는 문제가 발견되었다. 대표적으로 조류 및 야생동물 통제와 관련된 위해요인의 경우 OP048(공항 조류 및 야생동물 생물학적 환경 서식지 관리, 활동 관리), OP024(운항 조류 및 야생동물 생물학적 환경 조류/야생동물 통제), OP105(공항 조류, 야생동물 통제 미흡 생물학적 환경 관리 미흡)와 같이 다수의 코드가 중복적으로 존재한다. 이처럼 표준코드 자체에 잘못되거나 모호한 부분이 있음에도 불구하고 본 연구는 규정에 따른 공식 기준을 임의로 조정할 수 없어 해당 코드를 그대로 분석에 적용해야만 했다. 이러한 문제는 LLM뿐만 아니라 인간 전문가의 분류 일관성까지 저해하는 원인이 되므로 표준코드 기준을 체계적으로 재정비할 필요가 있다.

셋째, 상황 인식(situational awareness)과 같은 인적요인(human factor) 항목이 지닌 개념적 포괄성이 LLM의 다중라벨 분류에 직접적인 영향을 미쳤다는 점이다. 항공안전관리의 특성상 거의 모든 기술적·운영적 결함의 근본원인을 역추적하면 결국 정보누락, 인지실패 등 작업자의 상황인식 오류로 귀결될 수 있다. 자유서술형 텍스트에 내포된 복잡한 문맥과 근본원인을 추론하는 데 능한 LLM은 이러한 연결고리를 논리적으로 포착하여 표면적인 1차 위해요인(시설 결함, 기상 악화 등) 외에도 인적(H) 카테고리 코드를 빈번하게 동시 부여하는 경향을 보였다. 이는 LLM의 단순한 오분류라기보다는 사실상 모든 위해요인의 원인으로 확장될 수 있는 상황인식이라는 범주의 광범위함이 LLM의 추론 과정에서 그대로 투영된 결과로 해석할 수 있다.

Ⅵ. 결 론

본 연구는 인력과 전문성이 부족한 소규모 항공 조직 및 민간 전문교육기관의 환경을 고려하여 비정형 항공안전 자율보고서의 위해요인(hazard) 표준코드 분류를 자동화하기 위해 검색증강생성(RAG) 기반 대형언어모델(LLM)의 적용 가능성과 정확도를 검증하였다. 이를 위해 50건의 자율보고 사례를 대상으로 인간 전문가의 분류 결과를 기준 정답으로 설정하고 4종의 최신 LLM인 ChatGPT 5.2 thinking, Gemini 3.1 Pro, NotebookLM, Google AI Studio(Gemini 3 Flash)의 다중라벨 분류성능을 비교 분석하였다.

연구 결과, NotebookLM이 80.0%의 가장 높은 정확도(accuracy)와 56.8%의 F1-score를 달성하며 가장 균형 잡힌 분류성능을 보였다. 카테고리 예측 편향성 측면에서는 NotebookLM과 Gemini 3.1 Pro가 인간 전문가와 가장 유사한 분포를 보인 반면 ChatGPT 5.2 thinking 모델은 존재하지 않는 OR(조직) 카테고리를 생성하는 등 코드를 과잉생성하고 카테고리를 넓게 확장하는 편향이 확인되었다.

이러한 결과는 LLM과 검색증강생성(RAG) 기반 위해요인 코드 분류가 SMS 자율보고 데이터 처리에서 충분한 활용 가능성을 지니고 있음을 알 수 있다. 특히 안전관리 전담인력이 부족한 소규모 항공조직이나 민간 전문교육기관에서는 자율보고서의 1차 분류단계에서 실무적 효용이 클 것으로 판단된다. 그러나 현 시점에서 LLM이 인간 전문가 판단을 완전히 대체하기에는 아직 한계가 있다.

본 연구 결과는 LLM 기반 위해요인 자동 분류의 가능성을 확인하는 동시에 현행 한국형 위해요인 분류체계 자체의 구조적 한계도 함께 드러냈다. LLM 분석 과정에서 규정상 임의로 조정할 수 없는 현행 표준코드 체계의 문제점이 일부 확인되었다. 예를 들어 조류 및 야생동물 통제와 관련된 위해요인 코드가 OP048, OP024, OP105 등 여러 항목으로 중복되어 존재하는 등 분류 체계의 일관성 부족 문제가 관찰되었다. 이러한 문제는 LLM 기반 자동 분류과정에서 정확도와 일관성을 저하시킬 수 있다. 따라서 향후 LLM 기반 위험요인 분석 시스템의 정확도를 더욱 향상시키고 실제 현장 적용성을 높이기 위해서는 현행 위해요인 분류체계 자체의 중복성과 구조적 불일치를 정비하는 작업이 병행될 필요가 있다.

결론적으로 본 연구는 항공안전데이터 처리 및 활용 과정에서 생성형 AI 기술을 적용할 수 있는 가능성을 실증적으로 확인하였다는 점에서 의의를 가진다. 따라서 실제 현장에서 적용 시 검색증강생성(RAG) 기반 모델을 전문가 보조 도구로 활용하되 최종 분류결과에 대해서는 항공안전 전문가들의 검토과정을 거치는 병행체계로 유지하는 것이 바람직할 것이다.

References

1. ICAO, "Accident/Incident Data Reporting (ADREP) Taxonomy", ICAO, 2026, Available from: https://www.icao.int/safety/AIG/taxonomy
2. Kim, J. H., Lim, J. J., and Lee, J. R., "A study on the structural analysis of aviation safety data and standard classification system", Journal of the Korean Society for Aviation and Aeronautics, 28(4), 2020, pp.89-101.
3. NASA, "ASRS Report Processing Workflow", NASA ASRS Technical Report, Moffett Field, CA, 2023.
4. Zhou, D., Zhuang, X., Cai, J., Zuo, H., Zhao, X., and Xiang, J., "An ensemble model using temporal convolution and dual attention gated recurrent unit to analyze risk of civil aircraft", Expert Systems with Applications, 236, 2024, p.121423.

5. Park, Y. R., Kim, J. H., Choi, H. S., and Jeong, M. J., "A study on the sharing and utilization of domestic aviation safety information based on FAA cases", Journal of Advanced Navigation Technology, 26(2), 2022, pp.54-62.
6. Ministry of Land, Infrastructure and Transport, "Regulations on the processing and utilization of aviation safety data", MOLIT Instruction No. 1806, Sejong, 2024.
7. Tanguy, L., Tulechki, N., Urieli, A., Hermann, E., and Raynal, C., "Natural language processing for aviation safety reports: From classification to interactive analysis", Computers in Industry, 78, 2016, pp.80-95.
8. Yang, S. H., Choi, Y., Jung, S. Y., and Ahn, J. H., "A study on automatic classification of risk-based aviation safety data using natural language processing algorithms", Journal of Advanced Navigation Technology, 26(6), 2022, pp.528-535.
9. Devlin, J., Chang, M. W., Lee, K., and Toutanova, K., "Bert: Pre-training of deep bidirectional transformers for language understanding", Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, 2019, pp.4171-4186.
10. Nanyonga, A., Joiner, K. F., Turhan, U., and Wild, G., "Deep learning approaches for classifying aviation safety incidents: Evidence from Australian data", AI, 6(10), 2025, p.251.
11. Basil, E., "Large language models for aviation safety: Enhancing incident analysis through summarization of ASRS reports", M.S. Scholarly Paper, University of Maryland, College Park, 2025.
12. Ahmadi, A., Sharif, S., and Banad, Y., "Improving aviation safety analysis: Automated HFACS classification using reinforcement learning with group relative policy optimization", arXiv preprint arXiv:2508.21201, 2025, Available from: https://arxiv.org/abs/2508.21201
13. Tonmoy, S. M. T. I., Zaman, S. M., Jain, V., Rani, A., Rawte, V., Chadha, A., and Das, A., "A comprehensive survey of hallucination mitigation techniques in large language models", arXiv preprint arXiv:2401.01313, 2024, Available from: https://arxiv.org/abs/2401.01313
14. Bernardi, M. L., Cimitile, M., Panella, G., Pecori, R., and Simoncelli, G., "Automatic generation of job safety reports with explainable RAG-based LLMs", Information Systems Frontiers, 2025, pp.1-15.
15. OpenAI, "File Search - Assistants API", OpenAI Platform Documentation, 2024, Available from: https://platform.openai.com/docs/assistants/tools/file-search
16. Google, "File Search | Gemini API", Google AI for Developers Documentation, 2025, Available from: https://ai.google.dev/gemini-api/docs/file-search