

RAG 기반의 갈등 예측과 해소를 위한 추천 문장 생성

(Generating Recommended sentences for Conflict Prediction and Resolution based on RAG)

김진수

(Jin Su Kim)

요약

최근에 팀 프로젝트가 일반적인 업무수행 방식으로 자리 잡으며, 메신저를 통한 의사소통에서 감정 충돌로 인한 갈등이 자주 발생하고 있다. 이러한 갈등은 팀워크를 약화시키고 학습 및 프로젝트 성과에 부정적인 영향을 줄 수 있다. 본 연구에서는 팀 프로젝트 대화에서 갈등을 조기에 감지하고 예방하기 위해, 감정 예측 기반의 RAG(Retrieval-Augmented Generation) 기술을 이용하여 갈등 해소를 위한 추천 문장을 생성한다. 팀원 간의 대화를 감정 모델을 통해 먼저 감정을 예측하고 긍정/부정의 감성으로 분류하고, 벡터로 변환된 대화 간의 의미 유사도를 통합하여 갈등상태를 예측한다. 최종적으로 예측된 갈등상태, 예측 감정, 대화 문장으로 갈등 해소에 도움을 줄 수 있는 추천 문장을 생성한다. 생성된 추천 문장은 갈등 해소에 도움을 주는 일반적인 문장 뿐만 아니라 로컬 데이터베이스에 저장된 특화된 정보 파일도 검색함으로써 특정 영역까지 확대하여 갈등 해소를 위한 효율적인 추천 문장을 생성할 수 있다.

■ 중심어 : 검색 증강 생성 ; 갈등 ; 문장 추천 ; 감정 ; 문서 유사도

Abstract

Recently, as team projects have become a common way of performing work, conflicts due to emotional conflicts in communication via messengers have frequently occurred. Such conflicts can weaken teamwork and have a negative impact on learning and project performance. In this study, in order to detect and prevent conflicts in team project conversations early, we generate recommended sentences for conflict resolution using the RAG(Retrieval-Augmented Generation) technique based on emotion prediction. The conversations between team members are first predicted using an emotion model, classified into positive/negative emotions, and the conflict status is predicted by integrating the semantic similarity between the conversations converted into vectors. Finally, recommended sentences that can help resolve conflicts are generated using the predicted conflict status, predicted emotions, and conversation sentences. The generated recommended sentences can be expanded to specific areas by searching not only general sentences that help resolve conflicts, but also specialized information files stored in a local database, thereby generating efficient recommended sentences for conflict resolution.

■ keywords : Retrieval-Augmented Generation ; Conflict, Sentence Recommendation ; Emotion ; Sentence Similarity

I. 서론

최근에 팀 프로젝트 내 협업이 필수적인 업무 수행 방식으로 자리 잡고 있어서 다양한 개인의 감정이 충돌하여 갈등이 발생하는 상황도 빈번

해지고 있다. 특히, 인터넷 기술의 발달로 인하여 대면이 아닌 비대면 방식의 메신저를 통한 대화의 경우에 갈등이 빈번하게 발생한다. 갈등은 때때로 창의적 결과를 이끌어낼 수 있지만, 관리되지 않은 감정은 팀워크를 약화하고 프로젝트 성

* 정회원, 안양대학교 아리교양대학

과를 저하할 위험이 있다[1]. 또한, 프로젝트를 수행하는 과정에서 의사소통의 어려움과 갈등이 발생할 수 있으며, 이는 학습 성과에 부정적인 영향을 미칠 우려가 있다. 특히, 팀 프로젝트 기반 학습에서는 학생들이 경험하는 갈등의 상당 부분이 팀원 간의 관계 문제나 소통 과정에서 발생하는 부정적인 감정과 밀접한 연관이 있다[2]. 이에 따라 감정을 정확히 예측하고, 상황에 맞는 적절한 대응 문장을 제시하는 기술의 필요성이 대두되고 있다. 특히, 검색 기반과 생성 기반의 장점을 결합한 RAG(Retrieval-Augmented Generation) 기술로 감정 상태를 고려한 공감적이고 중재적인 문장을 추천하면 갈등을 효과적으로 해결하고 최소화할 수 있다.

본 연구는 프로젝트팀원 간의 대화 내용을 분석하여 갈등을 조기에 감지하고 예방할 수 있는 시스템 개발을 목표로 하고 있다. 즉, 감정 예측을 기반으로 RAG를 적용하여 팀 내 갈등을 예방하고, 건강하고 적절한 소통을 지원하는 문장을 추천하여 갈등을 최소화하고자 한다.

II. 관련 연구

1. 감정과 갈등의 관계

자신의 감정 상태에 따라 같은 상황도 다르게 반응할 수 있다. 즉, 스트레스, 피로, 불안과 같은 부정적인 감정 상태에서 상대방의 말이나 행동은 더 쉽게 위협적으로 받아들여지거나 공격적으로 반응할 가능성이 높아 갈등으로 비화될 수 있다[3]. 기본 감정은 기쁨(happiness), 분노(anger), 혐오(disgust), 공포(fear), 슬픔(sadness), 놀람(surprise)으로 Paul Ekman이 정의하였고, 감정은 남녀노소, 지역, 문화 등과 상관없이 누구나 일반적으로 느낄 수 있다[4]. 개인의 감정 인식 및 조절 능력인 정서 지능은 갈등 상황에서 더 성숙한 대응을 가능하게 한다. 즉, 감정 조절 능력이 높은 사람은 충동적으로 반응

하지 않고, 갈등 상황에서도 상대의 감정을 이해하고 효과적으로 소통할 수 있어 더 협력적인 방식으로 갈등을 해결할 수 있다[5].

갈등은 충돌, 불일치, 부조화, 경쟁, 논쟁, 싸움 등과 같은 비양립적 구조를 띠며, 이러한 갈등 상황은 구성원들에게 불안, 긴장 등과 같은 부정적인 감정을 유발할 수 있다. 대인관계 맥락에서 갈등은 개인 간 성격의 불일치로 인해 개인과 팀 차원에서 불쾌한 감정이나 태도를 인지하는 상황으로 설명된다[2]. 또한, 갈등을 과업갈등과 관계갈등의 두 측면에서 접근할 때, 과업갈등은 반드시 부정적인 결과만을 초래하는 것이 아니라, 팀 내 상호작용을 촉진하고 전반적인 팀 성과에 긍정적인 영향을 미칠 수 있는 잠재력을 지닌다[6]. 반면, 관계갈등은 팀 구성원 간 개인적 특성의 차이에서 기인한 부정적인 감정으로 인해 발생하며, 이는 구성원 간 불화를 유발하고 과업수행에 대한 동기를 저하하는 결과를 초래한다[7]. 또한, 분노, 적개심, 좌절감, 스트레스 등과 같은 개인적 감정은 팀 내 상호작용의 질을 저하시킨다[8]. 팀 내 갈등은 상황적 조건보다는 감정적 요인에 의해 더욱 쉽게 유발되기에 차별, 공격, 비난, 불쾌감을 유발하는 부정적인 언행을 사전에 인식하고 조율함으로써, 정서적 갈등을 예방하고 팀의 갈등을 최소화하는 전략적 접근이 필요하다.

2. 생성형 인공지능

생성형 인공지능(Generative AI)은 기계가 주어진 텍스트나 이미지 등의 데이터를 단순히 분석하는 수준을 넘어, 새로운 데이터를 스스로 생성하는 능력을 지향하는 기술로 정의된다. 해당 개념은 오래전부터 존재해 왔으나, 최근 딥러닝 기술과 LLM(Large Language Model)의 급격한 발전으로 인해 그 중요성과 활용 가능성이 크게 부각되고 있다. 생성형 AI의 대표적인 예로는 GPT(Generative Pre-trained Transformer) 계

열이 있으며, 이는 사전 학습(pre-training)과 미세 조정(fine-tuning)을 통해 뛰어난 자연어 생성 능력을 발휘한다[9]. 반면, BERT(Bidirectional Encoder Representations from Transformers)는 양방향 인코더 구조를 기반으로 하여, 자연어 생성보다는 분류, 추론, 문장 유사도 측정 등의 자연어 이해 작업에 강점을 보인다[10]. 최근에는 이러한 BERT 기반 모델을 생성형 과제에 적용하기 위한 다양한 변형 모델 및 기법이 활발히 연구되고 있다.

생성형 AI의 급속한 발전을 견인한 핵심 동력은 초거대 언어 모델의 등장이다. GPT-3와 GPT-4와 같은 LLM은 수십억~수천억 개의 파라미터와 방대한 양의 학습 데이터를 기반으로, 이전 세대 모델을 능가하는 수준의 텍스트 이해 및 생성 능력을 확보하였다. 이러한 모델들은 번역, 요약, 콘텐츠 생성, 교육용 챗봇 등 다양한 분야에서 폭넓게 활용되며, 자연어 처리 기술의 새로운 지평을 열고 있다. 그러나 이와 같은 고성능 모델들도 여전히 한계점을 지니고 있으며, 대표적인 문제로는 환각(hallucination) 현상과 정보의 최신성 부족이 있다. 생성형 모델은 문법적으로 완성도 높은 텍스트를 생성할 수 있으나, 실제와 어긋나는 정보를 혼합하여 출력하는 현상을 피하기 어렵다. 또한, 학습된 시점 이후의 새로운 정보에 대한 반영이 어려워, 빠르게 변화하는 정보 환경에서는 사실 기반 응답의 부족이라는 구조적 한계가 존재한다. 아울러, 해당 모델이 어떠한 과정을 거쳐 특정 답변을 생성했는지를 설명하기 어려운 비투명성 문제도 지적되고 있다.

GPT와 BERT는 자연어 처리 분야에서 가장 대표적인 두 가지 모델로 평가된다. GPT는 트랜스포머의 디코더 구조를 활용하여 왼쪽에서 오른쪽으로 단어를 예측하며 학습하며, 주로 문장 생성, 요약, 번역 등 텍스트 생성 중심의 작업에 효과적이다. GPT는 문맥을 고려하여 자연스럽고 유창한 문장을 생성할 수 있으며, 사전 학습

된 모델을 기반으로 Zero-shot 또는 Few-shot learning을 통해 다양한 작업에 빠르게 적용될 수 있다는 장점이 있다. 반면, 환각 현상, 높은 계산 비용, 정보 기반 응답 생성에 상대적으로 약하다는 단점도 존재한다. BERT는 트랜스포머 인코더 구조를 활용하여 문장을 양방향으로 학습하고 이해하는 데 중점을 둔 모델이다. 문장의 전후 문맥을 모두 고려할 수 있어 질의응답, 문장 분류, 개체명 인식 등 다양한 자연어 이해 과제에서 높은 정확도를 보인다. 또한, 사전 학습 이후 도메인 특화 데이터로 미세 조정이 가능하며 특정 과제에서 우수한 성능을 낼 수 있다. 다만, 자연스러운 텍스트 생성을 위한 구조는 아니며, 입력 길이 제한으로 인해 긴 문서의 전체 맥락을 유지하는 데 한계가 있다. 또한 특정 작업에 대한 추가적인 미세 조정 과정이 요구된다. 즉, 텍스트 생성 중심의 과제에는 GPT 계열이 효과적이며, 정확한 문맥 이해와 분석이 요구되는 과제에는 BERT 계열이 더욱 적합하다. 따라서, BERT 기반의 검색 기능과 GPT 기반의 생성 기능을 결합하여, 정보의 정확성과 생성 응답의 자연스러움을 동시에 확보하여 효율을 극대화할 수 있는 방식도 활용된다.

KoBERT는 BERT의 트랜스포머 인코더 구조를 그대로 유지하면서, 한국어에 특화된 성능을 개선하기 위해 개발된 언어 모델이다. 해당 모델은 한국어 성능 향상을 위해 위키백과, 뉴스 등 다양한 한글 문서를 수집하고, 이로부터 구성된 대규모 한국어 말뭉치를 기반으로 학습한다[11]. KoBERT는 특히 한국어의 불규칙한 어형 변화 특성을 효과적으로 반영하기 위해 데이터 기반 토큰나이제이션 기법을 도입하여 사용되는 토큰 수를 절감하면서도 성능 향상이라는 긍정적인 결과를 도출한다[12].

3. RAG(Retrieval Augmented Generation)

기존의 LLM은 특정 시점에 수집된 데이터로

학습되기 때문에, 실제로 존재하지 않는 정보를 생성하는 환각 현상 문제를 보완하기 위해 고안된 개념이 RAG이다. RAG는 실시간으로 외부의 최신 정보를 검색하고, 이를 생성 과정에 반영함으로써 응답의 정확성과 신뢰성을 향상하는 방식을 채택한다. 단순한 LLM 기반의 응답 생성보다 RAG를 적용할 경우, 보다 정확하고 최신의 정보를 제공할 수 있으며, 도메인 특화 지식의 활용, 비용 절감, 모델 자체 한계의 보완 등 다양한 이점을 확보할 수 있다.

RAG는 검색과 생성을 결합한 인공지능 기술로 검색 모듈, 강화 문맥 모듈, 생성 모듈로 구성된다[13]. 첫째, 검색 모듈은 외부 데이터로부터 필요한 정보를 추출하는 역할을 한다. 학습 데이터에만 의존하는 경우 지식의 최신성과 정확성이 저하될 수 있으므로, 방대한 문서 집합이나 데이터베이스로부터 실시간으로 자료를 검색하여 정보를 보완한다. 일반적으로 TF-IDF와 같은 전통적인 통계 기반 검색 알고리즘이나 문장 또는 단어를 임베딩하여 벡터 간 유사도를 계산하는 임베딩 기반 검색 방법이 사용된다. 둘째, 강화 문맥 모듈은 검색된 정보를 요약·추출·필터링하여 생성 모듈이 활용할 수 있는 간결하고 효율적인 문맥 정보를 제공한다. 이 과정은 프롬프트 엔지니어링과 밀접하게 연관되며, 문맥의 질과 관련성은 최종 생성 결과의 품질을 결정짓는 중요한 요소가 된다. 셋째, 생성 모듈은 강화 문맥 모듈에서 제공된 정보를 기반으로 최종 응답을 생성한다. 이 모듈은 외부로부터 획득한 정보와 내부적으로 학습된 지식을 종합하여, 사실성과 맥락적 일관성을 갖춘 텍스트를 생성하도록 설계되어 RAG 시스템 내의 각 모듈은 독립적으로 기능하면서도 상호 유기적으로 연결되어야 최고의 효율을 발휘할 수 있다.

III. 갈등 해소를 위한 추천 문장 생성

본 연구에서는 프로젝트팀원 간에 발생할 수

있는 갈등을 예측하여 갈등을 해소하거나 최소화하는 문장을 생성하여 추천하고자 한다. 먼저, 팀원 간의 정보를 주고받는 과정에서 표현된 문장의 감정과 대화 문장 간의 의미 유사도를 측정하여 갈등 상황을 예측한다. 대화 문장, 예측된 감정, 그리고 갈등 상황을 포함한 정보를 생성형 AI에 전달한다. 예측한 상황이 갈등상태가 아니라면 대화에 맞는 적절한 문장을 추천하고, 만일 갈등인 상황이라면 갈등을 최소화하거나 해결에 도움을 줄 수 있는 추천 문장을 생성한다. 이러한 상황에서, 프로젝트의 특성에 따라 LLM이 추천하는 문장이 실제 사용자에게 적합하지 않은 경우가 발생할 수 있다. 이는 특정 프로젝트에서 요구하는 조건, 수행 방법, 결과 지표, 긴급 변경 사항 등의 표현 방식이 일반적인 문장 구성과 다르거나, 해당 프로젝트만의 고유한 결정 사항이 존재하기 때문이다. 이러한 맥락에서, LLM이 사전 학습한 일반적인 지식은 오히려 부적절하거나 비효율적인 결과를 초래할 수 있다. 따라서 이러한 문제를 해결하기 위해 전체 프로젝트의 맥락을 충분히 반영하거나, 특정 집단 또는 조직 내에서 축적된 정보가 저장된 로컬 데이터베이스로부터 데이터를 추출하고 이를 학습한 후, 사용자 맞춤형 문장을 생성하여 추천한다.

수집된 프로젝트팀원 간의 대화 문장을 KoBERT의 입력으로 사용하여 중립 감정을 포함한 7가지 감정을 예측한다. 이러한 감정은 긍정적 감정(기쁨, 놀람, 중립)과 부정적 감정(슬픔, 공포, 분노, 혐오)으로 분류한다. 부정적인 감정은 상대에 대한 논쟁, 불신, 오해 등의 의미를 담는 경우가 많아 갈등의 원인이 되고 대화 문장 간의 내용이 상반되거나 문장의 의미가 반대인 경우에도 갈등이 발생할 수 있다.

따라서 감정과 문장 의미 유사도를 통해 갈등을 예측한다.

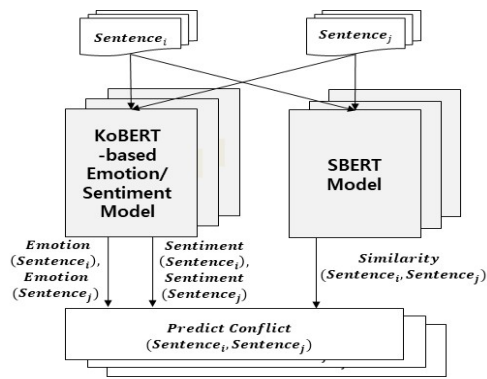


그림 1. 갈등 예측 시스템의 구조

그림 1은 프로젝트팀원들 간의 소통하는 텍스트를 입력하였을 때, 예측되는 텍스트 감정 및 감성과 텍스트 간의 의미 유사도를 판별하여 최종적인 갈등 상황을 예측하는 과정을 보여준다. 먼저 추출된 감정으로 긍정적 및 부정적 감성을 판별한다. 그리고 대화 문장 간의 의미 유사도는 SBERT(Sentence-BERT)를 통해 각 문장을 벡터로 변환한 후, 코사인 계수로 문장 간 유사도를 계산한다[14]. 즉, V_1 와 V_2 는 각 문장 $Sentence_i$ 와 $Sentence_j$ 의 벡터를 의미하고, numpy 함수로 $\text{dot}(V_1, V_2) / (\text{norm}(V_1) * \text{norm}(V_2))$ 코드로 구현한다.

문장에 부정어가 포함되었다면 긍정을 부정으로, 부정을 긍정으로 변환하는 절차를 거친다. 이때, 한국어 단형부정인 ‘안-’ 부정문과 장형부정인 ‘-지 않다’ 부정을 적용한다. 최종적인 갈등 예측은 문장 간의 의미 유사도와 대화 감성의 종류인 긍정-긍정, 긍정-부정, 부정-긍정, 부정-부정인 경우를 분류하여 예측한다. 이때, 문장 간의 의미 유사도인 코사인 계수 값이 0.3 이상이면 긍정(True), 0.3 미만이면 부정(False)으로 설정한다.

최종적으로 갈등 판별은 대화 문단의 감성이 긍정-긍정(AB)일 때는 유사도가 부정($\bar{C}$), 감성이 긍정-부정($A\bar{B}$)과 부정-긍정($\bar{A}B$)일 때는 유사도가 긍정(C), 그리고 문장의 감성이 모두 부정인 경우에는 갈등($Z = \bar{A}(\bar{B} + C) + A(B \oplus C)$)로 예측한다. 즉, 대화의 감성이 부정-부정인 경

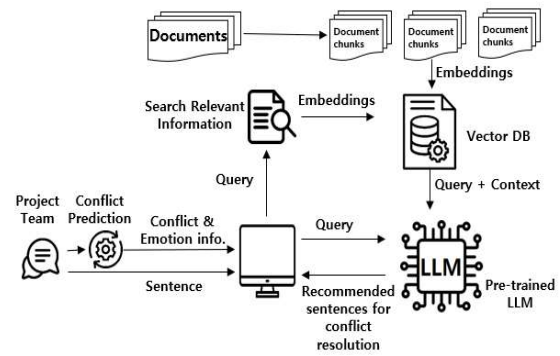


그림 2. 갈등 해결을 위한 추천 문장 생성 과정

우를 제외하고 갈등상태를 예측하기 위해 문장 간의 의미 유사도까지 고려한다.

그림 2는 갈등 최소화 및 갈등 해소를 위한 추천 문장을 생성하는 과정을 보여준다. 예측된 감정, 갈등상태, 그리고 팀 프로젝트 시 대화 내용을 취합하여 프롬프트 내용에 추가한 후, 갈등 유무에 따른 추천 문장을 생성한다. 개별화된 프로젝트 내에 갈등을 해결하거나 최소화할 수 있는 내용을 Vector DB로부터 추출한 적합한 텍스트를 사용하고, 그렇지 않다면 훈련된 대규모 언어 모델로부터 적절한 추천 문장을 생성하여 갈등을 해결할 수 있도록 추천한다.

IV. 실험 및 결과

본 연구에서 제안한 시스템은 갈등 예측을 위해 대화 내용에 표현된 감정을 예측하고 대화 문장 간의 유사도를 측정하여 최종적인 갈등을 예측한다. 그리고 대화문장, 대화 문장의 감정, 그리고 갈등상태를 포함한 정보와 RAG를 활용하여 갈등 해결 또는 최소화를 위한 문장을 생성하여 추천한다.

갈등 예측과 갈등상태에서의 적절한 추천 문장을 생성하기 위한 시스템 구성은 16GB의 Tesla T4 GPU, 2.2GHz Intel(R) Xeon(R) CPU, OS는 Ubuntu 22, 그리고 python 3.11 환경에서 Pandas, PyTorch, numpy 등의 라이브러리를 사용하였고 감정 데이터는 AI 허브에서 제공하는

감정 분류를 위한 대화 음성 데이터 집합의 텍스트만을 가져와 사용하였다[15]. 또한 갈등 예측을 위한 대화 데이터는 B라는 교과목을 수강하는 학생들을 대상으로 19개의 팀별 프로젝트 과정 중에 소통한 내용을 수집하였고, 이 중 88개의 갈등이 발생한 대화와 92개의 갈등이 없는 대화로 구성된다. 수집된 대화 내용 이외에도 프로젝트 관련된 정보를 가진 내용이 개별적인 파일에 저장되며 대부분 프로젝트 진행 중의 발생하는 일반적인 내용뿐만 아니라, 특정 프로젝트에 특화된 정확한 정보를 알고자 하는 내용도 포함한다.

표 1. 갈등 예측의 평가 결과(개)

		Predicted	
		Positive	Negative
Actual	Positive	81	7
	Negative	6	86

표 1은 전체 대화 내용을 바탕으로 감정과 문장 유사도를 적용하여 예측된 갈등상태와 비교한 결과이며, 성능 평가를 위해 혼동 행렬로 표현한 것이다. TP(True Positive), TN(True Negative), FP(False Positive), 그리고 FN(False Negative)에 속한 개수이며, 성능 평가에 사용된 정밀도(*Precision*, $TP/(TP+FP)$), 재현율(*Recall*, $TP/(TP+FN)$) 이외의 정확도(*Accuracy*)와 *F1-score* 값은 다음과 같다.

$$Accuracy = (TP + TN) / (TP + TN + FP + FN) \quad (1)$$

$$F1-score = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}} \quad (2)$$

*F1-score*는 수식에서 보인 것처럼 조화평균을 이용하기 때문에 산술평균을 이용한 것보다 큰 비중이 갖는 편향성을 줄여준다. 수식을 통해 계산된 정밀도, 재현율, 정확도, 그리고 *F1-score* 값은 약 93.10%, 92.05%, 92.78%, 그리고 92.57%로 비교적 높게 예측되었다. 비록 갈등상태를 잘못 예측한 FN과 FP의 예측 오류를 줄이기 위해서는 감정 예측의 정확도와 문장 간의 의미 유사

도를 높여야 한다. 그러나 뉘앙스와 같은 텍스트에 숨겨진 미묘한 감정이나 의미를 정확히 이해하는 것은 시간과 자원 문제 등의 제한을 갖는다.

표 2. 제안된 시스템의 갈등 해소 추천 문장 사례

대화 문장/감정/예측값	갈등 해소 추천 문장
A: 다른 숙제 때문에 진행을 하나도 하지 못했어!!! /슬픔/ [-1.71, 0.38, 1.48, 2.76, -1.65, 0.70, -1.63] B: 나도 못했어. 기한 후에 제출하면 감점되겠지? 어찌지? /공포/ [3.82, -2.89, 0.90, 2.77, 1.94, -2.05, -0.28]	(생성형 AI) 우리 서로 힘든 상황 이해하고, 함께 계획 세워 조금 늦더라도 최선을 다해 제출해보자
A: 오늘도 즐겁게 프로젝트를 합시다. 파이팅!!! /기쁨/ [-2.74, -1.15, 0.96, 0.31, -2.23, 2.49, 1.51] B: 나는 즐겁게 프로젝트를 마칠 수 없습니다. 코드가 쓰레기네요 /혐오/ [-1.65, -2.69, -0.42, 0.78, -1.41, 0.75, 4.36]	(생성형 AI) 서로 솔직한 마음을 존중하고, 부족한 부분은 함께 고쳐가면서 더 나은 결과를 만들어보자!
A: 오늘도 즐겁게 프로젝트를 합시다. 파이팅!!! /기쁨/ [-2.74, -1.15, 0.96, 0.31, -2.23, 2.49, 1.51] B: 그래요. 오늘 마무리 하고 꼭 쉬죠. ㅎㅎㅎ /기쁨/ [-2.74, -1.16, -1.40, 0.31, 0.97, 2.54, -1.22]	(생성형 AI) 서로 힘을 합쳐 즐겁게 프로젝트를 마무리하고, 좋은 결과를 함께 만들어보자!
A: 프로젝트 제출 기한과 프로젝트에 필요한 제약사항은 뭐죠? /공포/ [2.12, -3.30, 1.86, 2.72, -0.24, -1.62, 1.12] B: 전체 부분만 신경 쓰느라 못 들었는데? 직접 알아보면 안되나요? /혐오/ [0.16, 1.47, 0.43, -1.51, 1.32, -0.83, 1.78]	(생성형 AI) 서로 필요한 정보를 공유하면 모두에게 도움이 될 테니, 함께 확인하고 부족한 부분은 같이 채워나가자! (RAG 기반 생성형 AI) 프로젝트는 15주차 수업 전까지 제출하며, 블록, 주석, 복제, 변수, 리스트, 내 블록을 포함해야 합니다.

갈등 예측을 위한 사례 중, 표 2에 제시된 “다른 숙제 때문에 진행을 하나도 하지 않았어!!!”의 감정 예측은 공포, 놀람, 분노, 슬픔, 중립, 기

쁨, 혐오 예측값이 각각 [-1.82, 0.41, 1.53, 2.75, -1.55, 0.75, -1.73]이므로 슬픔이다. “나도 못했어. 기한 후에 제출하면 감점되겠지?”의 감정 예측값이 [3.82, -2.89, 0.90, 2.77, 1.94, -2.05, -0.28]이므로 공포이다. 두 대화의 감정이 슬픔과 공포인 부정적 감성이고, 두 문장 간의 유사도는 0.48로 0.3 이상으로 문장 간의 의미가 유사하므로 갈등이 발생할 가능성이 높게 예측된다. 또 다른 사례는 “오늘도 즐겁게 프로젝트를 합시다. 파이팅!!!”, [-2.74, -1.15, 0.96, 0.31, -2.23, 2.49, 1.51]와 “나는 즐겁게 프로젝트를 마칠 수 없습니다. 코드가 쓰레기네요”, [-1.65, -2.69, -0.42, 0.78, -1.41, 0.75, 4.36]의 대화의 감정은 각각 기쁨과 혐오이고 유사도가 0.34로 갈등이 발생할 가능성이 높다. 이러한 갈등상태가 높은 경우 갈등 해소를 위한 추천 문장을 생성하여 팀원 간의 갈등이 발생하지 않도록 도움을 주어야 한다.

일반적인 갈등 해소를 위한 추천 문장과는 다른 특정한 프로젝트 영역에 해당하는 프로젝트의 팀별 주제, 설명, 구현 범위, 제약사항, 제출 기한, 주차 별 진행 상황, 구현 시 참고 문헌, 이미지, 소리 등의 멀티미디어 자료의 유용한 사이트 등을 작성 및 수집하고 문서로 정리하여 벡터 공간에 임베딩한다. 즉, RAG는 LangChain을 통해 문서를 벡터로 임베딩하여 벡터 DB에 저장하고, 질의 시 유사 문서를 검색하여 LLM에 함께 전달하여 응답을 생성하도록 한다.

표 2는 프로젝트 진행 시 팀원의 대화에서 발생하는 갈등, 또는 특정 정보에 대한 질의 등이 포함되었을 때, RAG 기반의 생성형 AI로 문장을 생성하고 추천한 사례를 보여준다. 대화 내용이 갈등상태일 때 생성형 AI를 통해 생성된 추천 문장은 갈등 해소에 도움이 되었지만, 마지막 사례처럼 프로젝트 기한이나 제약사항과 같은 특정 상황에 대한 질의가 포함된 문장을 생성형 AI는 일반적인 추천 문장을 생성하지만, 대화자가 원하는 정확한 정보가 포함되지 않는 내용이다. 그러나 RAG 기반의 생성형 AI는 대화자가 입력

한 질의에 대한 가장 관련성이 높은 문서나 정보를 찾기 위해 로컬 데이터베이스에 저장된 내용을 검색함으로써 대화자가 찾거나 알고자 하는 적절한 문장을 추천해 주었다. 즉, 이러한 특정 범위에 속한 기본적인 생성형 AI가 알지 못하는 영역을 저장된 내용을 검색함으로써 프로젝트의 효율을 높일 수 있다. 비록, RAG 기반 생성형 AI는 최신 정보 반영이 가능하지만, 데이터베이스에 사용자가 검색하고자 하는 내용이 없거나 부족하다면 낮은 품질이나 잘못된 답변을 생성할 수 있다. 따라서 지속적인 피드백 처리를 통해 다양한 데이터를 저장하여야 한다.

V. 결론

본 논문에서는 팀별 프로젝트 진행할 때, 비대면 방식의 메신저를 통해 정보나 의견을 교환할 때 발생할 수 있는 갈등을 해소하기 위해 생성형 AI를 활용하여 문장을 추천하였다. 이를 위해 먼저 대화 내용의 감정을 예측하고, 예측한 감정을 긍정 및 부정적인 감성으로 판단한 후, 대화 문장 간의 문장 유사도를 적용하여 갈등상태를 예측한다. 그뿐만 아니라, 갈등이 발생할 가능성이 높은 경우 생성형 AI를 통해 갈등 해소 문장을 추천한다. 이때 특정한 프로젝트에서만 사용되는 내용이나 제약사항 등은 학습되지 않기 때문에 RAG를 통해 로컬 데이터베이스에 저장된 정보를 통해 제공하면 더욱 효율적으로 제공할 수 있음을 보였다. 즉, 프로젝트 진행 시의 학생들이 겪는 갈등 경험 중 팀원 간 관계 문제, 정보, 의사소통 시 발생하는 부정적인 감정과 문장 간의 유사도를 고려하여 갈등을 예측함으로써, 갈등을 해소하거나 최소화할 수 있는 추천 문장을 추천하여 다양한 팀원 간의 문제를 해결하는 데 기여할 수 있음을 보였다.

향후 연구에서는 텍스트로 작성된 대화 내용뿐 아니라 음성, 영상 등을 동시에 처리할 수 있는 멀티모달을 통해 정확도를 높이고, 낮은 비용과

적은 자원으로라도 갈등 예측 및 문장 추천을 수행할 수 있는 시스템을 고려해야 할 것이다.

REFERENCES

- [1] N. Y. Kwon, I. S. Oh, M. S. Kim, "Risk-Taking Orientation, Conflict, and Creativity: A Multilevel Perspective," *Korean Journal of Management*, vol. 27 no. 1, pp. 55-71, 2019.
- [2] S. B. Cho, "A Case Study of Undergraduate Student' Experience of Conflict in the Team-based Project Learning," *Ph.D dissertation, Sungkyunkwan University*, Aug. 2018.
- [3] M. Andrew, M. Colin, "Cognitive Vulnerability to Emotional Disorders," *Annual review of clinical psychology*, vol. 1, pp. 167-195, Apr. 2005.
- [4] K. R. Scherer, P. Ekman, Approaches to Emotion, Psychology Press, New York, 2014.
- [5] J. D. Mayer, P. Salovey, D. R. Caruso, "Emotional intelligence: Theory, findings, and implications," *Psychological Inquiry*, vol. 15, no. 3, pp. 197-215, 2004.
- [6] J. M. Lee, H. Y. Chung, "The Effects of Conflicts and Team Efficacy on Team Commitment and Team Achievement in University Team Project-Based Learning," *The Journal of Yeolin Education*, vol. 25, no. 4, pp. 269-287, 2017.
- [7] T. L. Simons, R. S. Peterson, "Task conflict and relationship conflict in top management teams: the pivotal role of intragroup trust," *Journal of Applied Psychology*, vol. 85, no. 1, pp. 102-111, 2000.
- [8] H. S. Choi, "The Effect of Task Conflict on Team Commitment and Team Performance: Moderating Effect of Self Efficacy and Team Efficacy," *Korean Journal of Business Administration(KJBA)*, vol. 25, no. 2, pp. 1019-1039, 2012.
- [9] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, Improving language understanding by generative pre-training, OpenAI Blog, 2018.
- [10] J. Devlin, M. W. Chang, K. Lee, K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *NAACL-HLT*, vol. 1, pp. 4171-4186, 2019.
- [11] SKTBrain, "Korean BERT pre-trained cased (K o B E R T) , " <https://github.com/SKTBrain/KoBERT>(accessed Feb., 1, 2025).
- [12] J. B. Lee, "Kcbert: Korean comments BERT," *Proc. of the 32nd Annual Conference on Human and Cognitive Language Technology*, pp. 437-440, 2020.
- [13] Y. C. Yoon, S. K. Kim, "Trends and Prospects of Retrieval-Augmented Generation (RAG) for Generative AI," *The Journal of Korean Association of Computer Education*, vol. 28, no. 2, pp. 69-80, 2025.
- [14] N. Reimers, 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 (EMNLP-IJCNLP)*, pp. 3982 - 3992, Hong Kong, China, 2019.
- [15] AI-Hub, <https://www.aihub.or.kr> (accessed Feb., 1, 2025).

저 자 소 개



김진수(정희원)

1998년 인천대학교 전자계산학과 학사 졸업.

2001년 인하대학교 컴퓨터공학과 석사 졸업.

2010년 인하대학교 정보공학과 박사 졸업.

2011년 ~ 현재 안양대학교 아리교양대학 부교수

<주관심분야 : 인공지능, 자연어처리, 딥러닝, 기계학습>