

저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

• 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건 을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 이용허락규약(Legal Code)을 이해하기 쉽게 요약한 것입니다.

Disclaimer 🖃





공학석사학위논문

반도체 설비의 지식 관리를 위한 SBERT 기반 조치 중심 요약 및 키워드 추출

SBERT based action focused summarization and keyword extraction for knowledge management in semiconductor facility

2023 년 2 월

서울대학교 대학원 산업공학과

한 누 리

반도체 설비의 지식 관리를 위한 SBERT 기반 조치 중심 요약 및 키워드 추출

SBERT based action focused summarization and keyword extraction for knowledge management in semiconductor facility

지도교수 조성 준

이 논문을 공학석사 학위논문으로 제출함 2022 년 11 월

서울대학교 대학원 산업공학과 한 누 리

한누리의 공학석사 학위논문을 인준함 2022 년 12 월

위 원 장 <u>장 우 진</u>(인)

부위원장 <u>조 성 준</u>(인)

위 원 <u>박건수</u>(인)

초록

설비 운영 기록 중 고장 조치 방법에 관한 내용은 유사한 고장 발생 시 참고하는

중요한 정보이다. 반도체 생산 환경에서 설비 운용 기록은 월 수만 건이 생산되고

있으므로 시스템에 의한 요약 없이는 조치 방법 분석이 실질적으로 불가능하지만,

기존 문서 요약 방식은 전반적인 정보를 요약하는 한계가 있다. 본 연구에서는

고장과의 관련 여부를 분류하도록 SBERT를 학습, 조치 방법을 중심으로 문서를

요약하고 반도체 용어 정보를 활용하여 요약본으로부터 핵심 키워드를 추출하는

프레임워크를 제안하였다. 이를 통해 기존 문서 요약 방식 대비 ROUGE 성능이

개선되는 효과를 얻었다. 본 연구 결과로 엔지니어가 필요한 정보를 얻는 속도를

높이고, 안정적으로 지식이 관리될 수 있도록 하여 생산성이 향상될 것으로

기대한다.

주요어: 추출 요약, 키워드 추출, 전처리

학번: 2021-21051

ii

목차

초특	루			i
목치	†			ii
丑	목치	†		iv
그림	빌	목차		v
제	1	장	서론	1
제	2	장	선행연구	8
		2.1 털	<u> 넥스트랭크</u>	9
		2.2 B	ERT	10
		2.3 B	ERTSUM	12
		2.4 S	BERT	13
		2.5 El	LECTRA ·····	15
		2.6 Si	imCSE	16
		2.7 D	Pecision-Focused Summarization	17
		2.8 번	반도체 제조 환경에서의 자연어처리	20
제	3	장	제안하는 방법론	23
		3.1 전	전처리	24

		3.2	분류기			 	 26
		3.3	최종 요약	및 키워	드 추출	 	 27
제	4	장	실험	결과			31
		4.1	실험 세팅	및 평가	지표	 	 31
		4.2 3	평가 결과	및 요약	예시	 	 32
제	5	장	결론				37
참고	1문	헌					39
Ab	stra	act					42

표 목차

丑	1.1	설비 운용 기록 예시	- 1
丑	2.1	추출 요약과 추상 요약의 예~~~~~	- 8
丑	3.1	전처리 과정	25
丑	3.2	분류기 학습 데이터 예시	27
丑	3.3	요약문에 포함된 문장 수에 따른 내용	28
丑	3.4	트라이그램 블록킹의 예	29
丑	3.5	반도체 축약어와 의미 예시	30
丑	4.1	시스템 요약문 평가 결과	33
丑	4.2	요약문 예시	34
丑	4.3	요약문 예시	35

그림 목차

그림	1.1	프로젝트 프레임워크	2
그림	1.2	검색어 관련 UI 디자인	4
그림	1.3	키워드 관련 UI 디자인	4
그림	1.4	예지보전	5
그림	1.5	예지보전의 효과	6
그림	2.1	BERT의 구조	- 10
그림	2.2	BERT의 사전 및 미세 학습·····	- 11
그림	2.3	BERTSUM의 구조·····	- 12
그림	2.4	BERTSUM의 요약층 구조	- 13
그림	2.5	문장 쌍 분류와 회귀 태스크	14
그림	2.6	ELECTRA의 구조······	15
그림	2.7	SimCSE의 비지도 학습 방법	16
그림	2.8	DecSum의 의사결정 대표성	18
그림	2.9	반도체 제조 환경에서의 AI 기반 지식 관리 시스템 개념도	- 20
그림	2.10	칩 지식 그래프	22
그림	3.1	제안하는 방법론	· 23
그림	3.2	조치 중심의 요약	26
그림	3.3	설비 운용 기록으로부터의 키워드 추출 과정 예시	- 30
그림	4.1	시스템 요약문 평가 결과	32

그림	4.2	키위드	추축	격과	예시	3	6
	1.4	´ I II —	12	2-7	11' '1	_	v

제 1 장 서론

반도체 설비 운용 기록은 설비에 문제가 발생했을 때 문제의 원인, 현황 그리고 조치 방법 등의 내용이 포함된 웹 문서를 의미한다. 설비 엔지니어는 운용 기록을 통해 상황을 공유하거나 유사한 고장을 다룬 기록을 참조하여 설비 상태를 개선하기 때문에 굉장히 중요한 데이터이다. 표 1.1은 실제 설비 운용 기록의 예이다. 설비 고장이 언제 발생했는지, 예상되는 원인과 실제 원인은 무엇인지, 그리고 어떻게 조치했는지 등 다양한 맥락의 내용이 있음을 알 수 있다.

Table 1.1: 설비 운용 기록 예시

구분	પીક
기록 1	Stack-1 Error발생 등 Hub fix시 Stack-1 Error가 발생함. Hub fix시 계측된 Hub MS Point의 Position이 Expected Position 대비 차이가 Threshold를 넘어감. 해당 Threshold는 설정에서 변경이 가능하나 통상적으로 10nm~20nm 정도임. 가능한 원인은 1. Hub screen 면 오염에 의한 Hub Weight 차이 2. Hub Side Tilt 등에 의한 Position Drift가 있음. Hub screen면 오염인 경우 특정 Hub에서 Trend가 상승하며, Side 의 Vacuum screen의 오염인 경우 전반적인 Trend가 상승한다. Hub Side, Tama Side 등 설비 구조적인 문제에서 유발되는 경우에도 전반적인 Trend가 상승한다. 본건은 전반적인 Trend가 상승한 경우로, HS Vacuum screen의 오염이나 TS, HS 등의 구조적인 문제로 의심할 수 있다. 1차적으로 Vacuum screen Clean을 실시하고, 개선이 없을 경우 DESO와 DESO에 따른 Recovery로 Focal 작업을 필요로 한다.
기록 2	MS side h2 drop에 의한 fixment fial. Side fix시 MS에 h2 drop이 있다고 판단되는 경우 이를 제거하기 위해 MS drop decay 작업을 진행. 1. 발생 원인 - MS side h2 drop에 의한 fixment fail - Robot 오염 - 2. 대책작업 Case 1 : 특정 MS side 에서만 발생하는 경우 - MS side cleaning. Case 2: MS side 구분 없이 발생 - Robot sony cleaning - Robot/MS side 오염으로 인해서 MS side fixment Point의 위치에 h2가 남겨짐. h2 drop 상태에서 MS fixment/Tilt를 계측하게 되면 low Tilt jump 됨. 0.7 이하일 경우 MS side를 RS side으로 이동하여 air shower를 이용하여 Dry.

운용 기록은 양식이 제한이 없어 내용을 자유롭게 서술하여 상황을 더욱 자세히 공유할 수 있는 장점이 있지만, 동시에 필요한 정보를 찾는 데 어려운 단점이 존재한다. 또한, 운용 기록들은 월 수만 건 생기기 때문에 참고할만한 기록들이 어디에 있는지 찾고, 그중에서도 설비 엔지니어가 가장 중요하게 생각하는 고장 조치 방법과 관련된 내용을 파악하는 데 시간이 소요되는 문제가 있다. 따라서 운용 기록을 고장 조치 방법을 중심으로 요약하는 것이 필요한데, 일반적인 요약 방식은 문서의 전반적인 맥락을 포함하기 때문에 한계가 있다. 더불어 설비 운용 기록들은 설비에 대한 노하우가 집약된 데이터이기 때문에 기록들로부터 정보들을 추출하고 검색될 수 있도록 하여 효과적으로 중요한 지식이 관리될 수 있어야 한다.

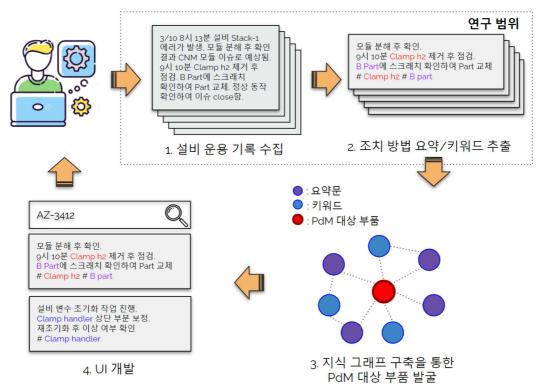


Figure 1.1: 프로젝트 프레임워크

본 연구는 설비 운용 기록을 고장 조치 방법을 중심으로 요약하고, 설비 축약어 데이터를 활용하여 키워드를 추출하는 방법론을 제안한다. 그림 1.1과 같이 설비 운용 기록을 수집하고, 조치 방법 중심의 요약과 키워드를 추출하고, 지식 그래프를 구축하여 예지보전(Predictive Maintenance, PdM) 대상 부품을 발굴, 최종적으로 사용자에게 UI를 제공하는 전체 프로젝트 중 요약과 키워드 추출과 관련된 부분이다. 문서의 각 문장을 고장 조치 여부와 관련되어 있는지 예측하고, 관련될 확률이 높은 상위 세 문장을 요약문으로 선택한다. 그 후 요약문과 설비 축약어 단어 간 겹치는 부분을 키워드로 선정하여 설비의 하드웨어 종류에 따라 군집화될 수 있도록 한다. 기존 문서 요약 방식 대비 요약 평가 지표인 ROUGE 점수가 약 0.2 정도 향상된 결과를 얻었다. 그리고 문서의 키워드 추출을 통해 그림 1.2, 1.3과 같이 설비 엔지니어가 필요한 정보를 효과적으로 얻을 수 있었다. 먼저 그림 1.2의 검색어 관련 UI는 설비의 고장 원인을 검색하고, 검색어와 관련된 고장 조치 요약문들을 시각화하여 사용자가 어떻게 조치할 것인지 의사결정을 도와주는 데 목적이 있다. 요약문을 클릭하게 되면 전체 텍스트가 팝업되어 보다 자세히 설비 운용 기록의 내용을 파악할 수 있다. 또한, 요약문들에 포함된 키워드들을 목록화하여, 고장 원인과 관련된 주요한 설비 부품들을 파악할 수 있게 한다. 키워드를 클릭하게 되면, 해당 키워드를 포함한 요약문들과 키워드 관계 그래프를 그림 1.3과 같이 볼 수 있다. 키워드 관계 그래프는 요약문과 키워드들을 노드로 하고, 요약문 안에 키워드가 포함되면 간선으로 연결된다. 그리고 매개 중심성(Betweenness Centrality)과 같이 네트워크의 중심 노드를 분석하는 척도와 네트워크 시각화를 활용하면, 설비 고장의 원인이 되는 중요한 부품들을 시각적으로 분석할 수 있게 된다. 이런

부품들을 우선으로 예지보전(Predictive Maintenance, PdM)의 대상으로 관리될 수 있도록 하여 설비 생산성을 향상할 수 있다.

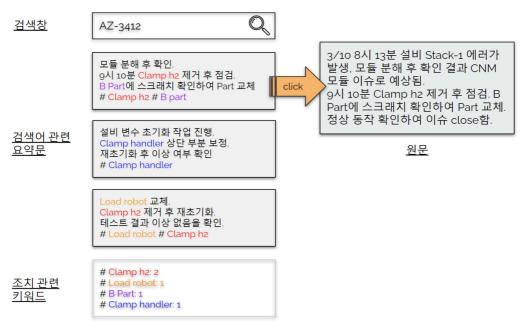


Figure 1.2: 검색어 관련 UI 디자인

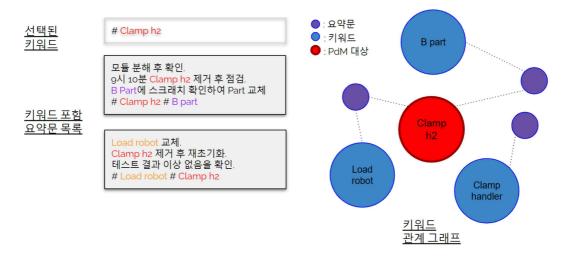


Figure 1.3: 키워드 관련 UI 디자인

예지보전을 위한 부품 발굴은 본 연구의 다음 목표로, 그림 1.4와 같이 센서값과 같이 설비의 건강 상태를 대변하는 데이터를 관찰하여 고장이 발생하기전에 미리 보수하는 활동을 말한다[1]. 예지보전의 방법 중 하나는 설비 부품의 상태를 대변하는 센서를 도메인 지식을 활용하여 이상값을 정의하고, 시계열 예측 알고리즘을 통해 이상값 도달하는 시점을 예측하는 것이다. 설비의 진동 데이터나, 온도 이미지, 동작 가능도와 같은 데이터들을 기반으로 시계열 예측을 통해 예지보전의 대상이 될 부품들을 추린다. 예지보전 모델은 예측 알고리즘을 통해서 정보를 가공하는데, 데이터의 경향을 분석하여 언제 설비가 개선되어야 하는지 판단한다. 단순히 설비의 부품이 고장 날 때까지 실행하게 되면 정상적으로 복구되는데 걸리는 시간이 오래 걸리고, 아직 사용 가능함에도 교체하게 되면 불필요한 비용이 증가하게 된다. 따라서 예지보전을 하게 되면 그림 1.5와 같이 설비의 생산성을 높이고, 보수 비용을 줄이는 효과를 얻게 된다.

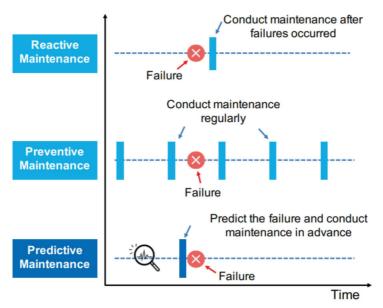


Figure 1.4: 예지보전

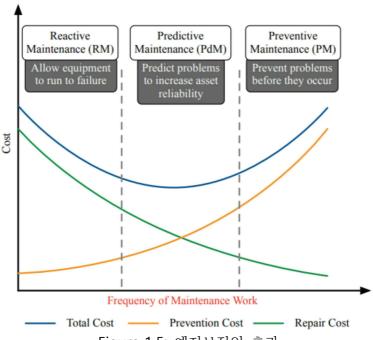


Figure 1.5: 예지보전의 효과

이런 장점에도 불구하고 예지보전만을 하지 못하는 이유는 정확한 예측을 하기 위해 많은 분석이 필요하기 때문이다. 많은 시뮬레이션과 과거 경험을 통해 부품의 이상치를 정의해야 하고, 부품의 건강 상태를 대변하는 센서가 존재해야 하며, 센서 데이터의 값이 이상치에 도달하기 전에 경향이 변하는 등 예측가능해야 한다. 갑자기 센서 데이터가 없거나 데이터가 이상치만큼 튀게 되면 예측할 수 있지 않다. 따라서 설비의 고장을 주로 발생시키는 주요한 부품들을 우선하여 가능한지 분석하여 예지보전의 대상으로 발굴되어야 한다. 이때 그림 1.3과 같은 그래프 UI를 활용하게 되면 부품들 간의 관계를 파악하고 문제가 많은 부품을 발굴하는 데 큰 도움이 될 것으로 기대한다.

본 논문의 구성은 다음과 같다. 2장에서는 텍스트 요약 기법 중 추출 요약(Extractive Summarization)과 문서를 임베딩 벡터로 만드는 관련 연구들을 살펴본다. 3장에서는 본 연구에서 사용하는 설비 운용 데이터의 특성, 전처리 방법 각 문장을 고장 조치 방법과 관련성을 고려하여 분류하는 방법, 최종적으로 의미적인 중복 없이 요약문을 선택하는 방법, 그리고 요약 결과물에서 키워드를 추출하는 방법에 대하여 자세히 설명한다. 4장에서는 실제 조치 방법을 중심의 요약문과 본 연구의 방법을 포함하여 문서 요약 모델의 요약문들을 ROUGE 점수와 요약된 결과물을 분석하여 평가한다. 5장에서는 본 연구의 결론과 의의에 대하여 논한다.

제 2 장 선행연구

문서 요약은 전체 문서의 전반적인 맥락을 담도록 요약하는 것을 의미한다. 인터넷의 발전으로 문서의 양이 폭발적으로 증가하고 있기에, 문서 요약은 중요한 태스크로 활발히 연구되고 있다. 문서 요약은 크게 추출 요약과 추상 요약으로 나뉘어 연구되고 있다. 먼저 추출 요약은 전체 문장 중 요약문에 포함될 문장들을 선택하는 방식이다

추상 요약은 문장을 선택하는 방식이 아니라, 새로운 요약문을 만드는 방식이다. 표 2.1은 추출 요약과 추상 요약의 예이다. 본 연구의 설비 운용 기록은 한글과 영어가 혼용되어 있고, 반도체 전문 용어가 대부분인 점을 고려하여 추출 요약 방식을 택하였다.

Table 2.1: 추출 요약과 추상 요약의 예

구분	<i>पा</i>
원문	3/10 8시 13분 설비 Stack-1 에러가 발생, 모듈 분해 후 확인 결과 CNM 모듈 이슈로 예상됨. 9시 10분 Clamp h2 제거 후 점검. B Part에 스크래치 확인하여 Part 교체. 정상 동작 확인하여 이슈 close함.
추출	9시 10분 Clamp h2 제거 후 점검. B Part에 스크래치 확인하여
요약	Part 교체
추상	Clamp h2 제거 후 점검 결과 B Part에 스크래치 확인하여
요약	새로운 부품으로 교체하였다.

2.1 텍스트랭크

텍스트랭크(TextRank)는 페이지랭크(PageRank) 알고리즘을 기반으로 한 추출 요약 방식[2]이다. 페이지랭크는 하이퍼링크를 가지는 웹 문서의 상대적 중요도에 따라 순서를 매기는 알고리즘으로, 참조되거나 유입되는 페이지가 많을수록 순위가 올라가는 방식이다. 페이지랭크는 어떤 사람이 어떤 특정한 페이지에 도달할 가능성을 대변한다. 페이지랭크는 웹 문서의 크기에 관계없이 계산될 수 있고, 이론적인 실제 값에 도달하기 위해 반복적인 연산이 필요하다. 식 (2.1) 은 N개의 웹 문서가 있을 때 페이지 A의 페이지랭크 값이다. 함수 L은 페이지가 가지고 있는 링크의 수를 의미하고, d는 다른 링크를 클릭하지 않고 해당 페이지에서 멈출 정도를 조절하는 댐핑 팩터(Damping Factor)이다.

$$PR(A) = \frac{1 - d}{N} + d(\frac{PR(B)}{L(B)} + \frac{PR(B)}{L(B)} + \frac{PR(B)}{L(B)} + \cdots)$$
 (2.1)

텍스트랭크는 페이지랭크에서의 각 웹 문서를 문장으로 대체한 알고리즘이다. 자세히 설명하자면, 문서의 문장을 노드로 하고, 문장 간 유사도를 간선으로 하는 그래프를 생성한다. 문장 간 유사도 값은 문장 벡터 간 코사인 유사도를 활용할수 있다. 그 후 페이지랭크 알고리즘을 적용하여 문장의 중요도를 측정하고, 요약문에 포함될 k개의 문장을 중요도가 높은 순서대로 선택하는 방식이다.

2.2 BERT

사전 학습된 언어 모델은 많은 양의 데이터로 임베딩 층을 학습하고, 다른 태스크에 맞게 사후 학습할 때 활용된다. 사전 학습된 언어 모델을 데이터의 양이 적은 태스크에 맞게 학습하게 되면 성능 향상을 기대할 수 있다. 따라서 사전 학습된 언어 모델은 문서 요약, 토픽 모델링 등 다양한 태스크가 존재하는 자연어처리에도 많이 연구되고 있다.

BERT(Bidirectional Encoder Representations from Transformers)는 트랜스포머(Transformer) 계열의 사전 학습된 언어 모델 중 하나이다[3]. 사람이 단어를 이해할 때 앞과 뒤 문맥을 고려한다. BERT는 사람처럼 문맥을 고려하여 단어를 임베딩하는 좋은 장점이 있다. 이런 장점이 있을 수 있는 것은 BERT가 그림 2.1과 같이 어텐션 기반의 여러 트랜스포머 인코더로 구성되어 있고, 또 마스크드 언어 모델과 다음 문장 예측과 같은 방식으로 사전 학습하기 때문이다.

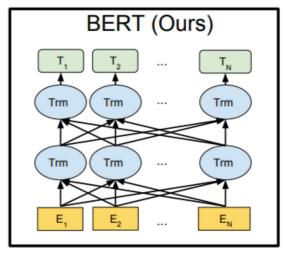


Figure 2.1: BERT의 구조

마스크드 언어 모델(Masked Language Model; MLM)은 입력 텍스트의 일부 단어를 무작위로 마스킹하고 이를 예측하는 방식이다. 입력 텍스트의 약 15%의 단어를 [MASK]로 변환하고, 나머지 단어들로 본래 어떤 단어였는지 예측하면서 문장을 이해한다. 예측하는 과정에서 [MASK] 전과 후의 단어들을 모두 활용하기 때문에, 양방향의 맥락을 이해할 수 있다.

다음 문장 예측(Next Sentence Prediction; NSP)는 주어진 두 개의 문장이 서로 이어지는 문장인지 예측하는 태스크이다. 이 태스크로 학습하는 이유는 두 문장의 관계를 이해하는 것은 질의응답(Question Answering, QA)과 같은 여러 자연어처리 태스크에 중요하기 때문이다. BERT에서는 문장 간 구분을 위해 [SEP]라는 특별한 토큰을 한다. 두 문장을 토큰으로 변환한 후에, [SEP]를 두 문장사이와 마지막에 붙여준다. 그리고 첫 번째 [CLS] 토큰이 입력되는 부분의 출력을 NSP 분류기로 입력하여 두 문장이 이어진 것인지 아닌지 학습하게 된다.

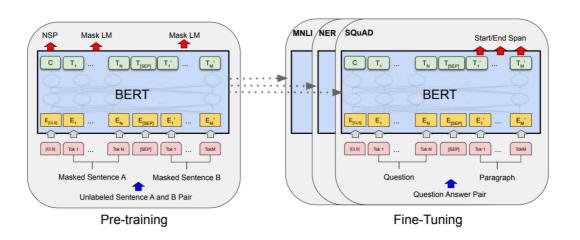


Figure 2.2: BERT의 사전 및 미세 학습

2.3 BERTSUM

BERTSUM은 그림 2.3와 같이 BERT를 통해 문장을 임베딩하고 요약층을 활용하여 각 문장의 요약문에 포함될 확률을 예측하는 모델이다[4]. BERT를 추출 요약 태스크에 활용하기 위하여 입력에 대한 수정과 요약층 추가와 같은 두 가지 개선을 하였다.

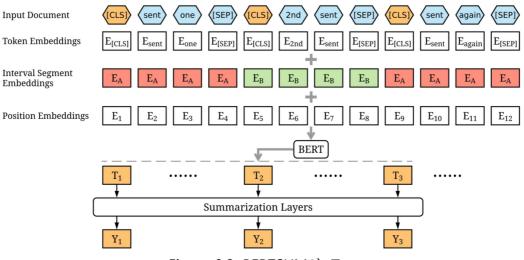


Figure 2.3: BERTSUM의 구조

BERT에서는 마스크드 언어 모델과 다음 문장 예측을 통해 사전 학습하기 때문에 하나 혹은 두 문장이 입력된다. 따라서 모델은 각 입력 토큰이 어떤 문장에 속하는지 알기 위해서 0 혹은 1의 Segment Embedding만 있으면 된다. 하지만 BERTSUM에서는 여러 문장이 입력될 수 있기에, 문장 앞뒤로 [CLS]와 [SEG] 토큰이 붙는다는 점과 Segment Embedding이 이어진 문장을 구분하기 위하여 홀수 번째 문장에 속한 토큰은 0, 짝수 번째 문장에 속한 토큰은 1이 붙는다. 각 문장을 BERTSUM에 입력되면 각 문장의 임베딩 벡터들이 출력된다. 임베딩 벡터들을 요약층에 넣으면 각 문장의 요약문에 포함될 확률들을 얻을 수

있다. 요약층으로 완전 연결 계층들로 구성된 간단한 분류기. 그림 2.4와 같이 두개의 트랜스포머 계층들을 얹은 분류기(Inter-sentence Transformer) 그리고 LSTM 기반 분류기를 실험한 결과 Inter-sentence Transformer를 활용한 경우가 가장 성능이 좋았다.

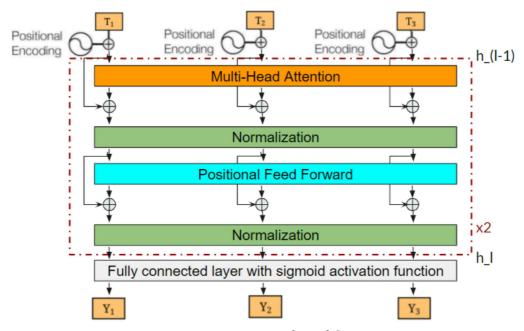


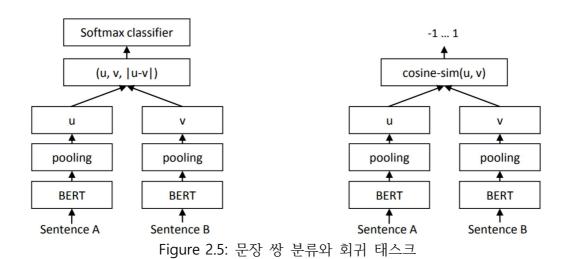
Figure 2.4: BERTSUM의 요약층 구조

2.4 SBERT

BERT는 마스크드 언어 모델과 다음 문장 예측과 같은 방법으로 비지도학습한다. 그렇기에 BERT의 출력 임베딩 벡터는 문장 군집화에 적합하지 않다. 다음 문장 예측 태스크로 학습할 때 모든 문장을 쌍으로 입력할 수 있긴 하지만, 문장의 개수가 n개이면 n(n-1)/2개의 비교를 해야 하기에 비효율적이다.

SBERT(Sentence BERT)는 BERT의 문장 임베딩을 성능을 개선한 모델이다[5].

문장을 대변하는 임베딩 벡터로 [CLS] 토큰의 출력, 평균 풀링 또는 최대 풀링한 출력을 쓴다. 그리고 그림 2.5와 같이 문장 쌍 분류 태스크와 문장 쌍 회귀 태스크로 BERT를 미세조정한다.



문장 쌍 분류 태스크에는 NLI(Natural Language Inferencing) 데이터 셋을 사용한다. NLI에는 두 문장을 주고 문장 간 관계가 수반, 모순 그리고 중립 중하나로 레이블 되어 있다. 첫 번째 문장의 출력을 u, 두 번째 문장의 출력을 v이라고 할 때, u와 v 그리고 u-v 세 가지 벡터를 이어서 소프트맥스 분류기에 입력하여 두 문장이 어떤 관계인지 학습한다. 문장 쌍 회귀 태스크에는 STS(Semantic Textual Similarity) 데이터 셋을 사용한다. STS에는 두 문장을 주고 문장 간 유사도가 0~5의 점수로 부여 된다. 그 후 u와 v 간 코사인 유사도를 계산하고, 데이터 셋의 유사도와의 평균 제곱 오차(Mean Squared Error; MSE)가 최소화되는 방향으로 학습된다.

2.5 ELECTRA

BERT의 마스크드 언어 모델은 입력의 약 15% 정도만을 마스킹하고, 다른 단어들로 마스킹 전 단어가 무엇이었는지 예측한다. 따라서 모델을 업데이트 하는데 데이터가 모두 활용되지 못한다. 데이터의 활용도를 높이기 위해 같은 문장을 여러 번 입력할 수 있겠지만, 학습 시간이 오래 걸리는 단점이 있다.

ELECTRA는 BERT가 안정적으로 성능을 내기 위하여 많은 양의 데이터와 학습시간이 필요한 단점을 개선한 모델이다[6]. 이를 위해 모든 토큰이 원래의토큰인지, 대체된 것인지 예측하는 사전 학습 방법인 대체 토큰 감지(Replaced Token Detection; RTD)를 제안한다. 그림 2.6와 같이 토큰을 대체하기 위해생성기(Generator)와 판별기(Discriminator) 구조를 사용한다. 생성기는 작은마스크드 언어 모델로 마스킹 된 토큰이 무엇인지 예측하는데, 원래 토큰과예측한 토큰이 같은지 혹은 다른지를 판별기가 예측하며 학습하게 된다. 학습이끝나면 판별기만을 사전 학습 모델로 활용한다.

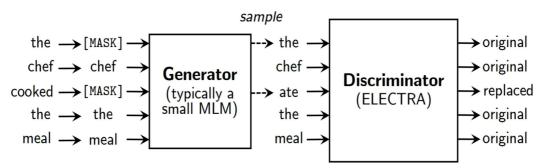


Figure 2.6: ELECTRA의 구조

2.6 SimCSE

SimCSE는 간단한 대조 학습(Contrasive Learning)을 통해 문장을 임베딩하는 프레임워크[7]이다. 대조 학습이란 특징 공간(Feature Space)상에서 특정 데이터를 기준으로 유사한 데이터는 가까이, 그렇지 않은 데이터는 멀리 위치하도록 학습해나가는 자기지도 학습(Self-supervised Learning) 방식이다. 데이터의 레이블을 다는 비용과 걸리는 시간이 많으므로, 레이블 없이도 지도 학습 방식과 유사하게 동작한다는 것은 큰 장점이 있다.

대조 학습을 함에 있어 중요한 점은 유사한 데이터를 만드는 방식이다. 이미지의 경우 자르거나, 회전하거나, 명암을 조절하는 등 다양한 방식으로 데이터를 증강할 수 있다. SimCLR은 드롭아웃(Dropout)을 노이즈로 사용하여 손쉽게 이미지 데이터를 증강하고 뛰어난 성능을 달성했다. SimCSE는 SimCLR의 아이디어를 텍스트에 적용하여 사전 학습한 모델이다. 트랜스포머 인코더에 드롭아웃을 적용하고 같은 문장을 두 번 입력한다. 이때 그림 2.7와 같이 같은 문장을 넣은 출력 쌍은 유사한 쌍으로, 나머지는 유사하지 않은 쌍으로 대조학습을 하게 된다.

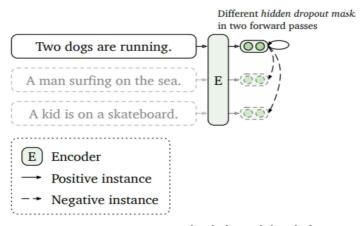


Figure 2.7: SimCSE의 비지도 학습 방법

2.7 Decision-Focused Summarization

사람의 의사결정 결정은 종종 수많은 문서를 읽고, 의사결정 결정에 필요한 정보들을 추리는 과정이 수반된다. 예를 들어 의학적인 판단을 하거나, 투자자가 주가의 추이를 예측할 때 관련된 사람, 기업들에 대한 정보에 대하여 분석이 필요하다. 이런 경우 문서 요약은 관련된 정보들로 줄여주기 때문에 의사결정 결정에 도움을 줄 수 있다[9].

일반적인 문서 요약은 특정한 의사결정 결정에 대한 견해 없이 전형적으로 텍스트의 정보에만 의존하여 정의된다. 그 결과, 암 진단 여부를 요약문을 통해 판단하고 싶은데, 무릎 부상과 같은 내용이 포함되는 경우처럼 의학 문서를 요약할 때 있어 어떠한 판단에 도움이 되지 않는 불필요한 정보가 포함될 수 있다. 하여 Hsu 등[10]은 의사결정에 집중된 요약이라는 새로운 문제를 제안한다. 의사결정에 집중된 요약의 목적은 의사결정과 관련된 정보들로 요약을 하는 것이다.

Hsu 등[10]의 모델(이하 DecSum)은 이를 위해 전체 텍스트에서 의사결정 결정을 예측하는 모델을 활용하였다. 예측 모델을 통해 텍스트로부터 어떻게 의사결정이 추론되는지 모델링하였다. 요약문을 만들기 위하여, 전체 텍스트로부터 예측한 의사결정 결정들과 유사한 의사결정 결정들을 도출할 수 있는 문장들을 선택하였다. 또, 요약문 간 의미상으로 중복되는 문장들이 포함되는 것을 피하고자 정량적으로 문장 간 유사성을 측정하여 요약문에 포함될 문장을 선택할 때 활용하였다. DecSum의 성능을 평가하기 위하여 Yelp 데이터 셋으로 레스토랑 평점 예측을 도와줄 수 있는 요약문을 만드는 실험을 진행하였다.

DecSum은 각 문장의 의사결정의 충실성과 대표성, 그리고 의미적 유사성을

정량화하여, 손실함수를 구성하고 반복마다 손실 값이 가장 크게 줄어들게 하는 문장을 요약문으로 선택한다. 단, 요약문의 첫 문장을 선택할 때는 의미적 유사성을 고려하지 않는다. 의사결정의 충실성은 전체 텍스트를 넣었을 때의 의사결정과 요약문을 넣었을 때의 의사결정이 유사해야 한다는 것이다. DecSum은 레스토랑 평가 텍스트를 입력, 평점을 출력으로 하는 예측 모델을 활용한다. 다시 말해, 전체 텍스트를 예측 모델에 넣었을 때와 요약문을 예측 모델에 넣었을 때 출력되는 평점의 값이 유사하다면, 요약문은 평점을 판단하는데 필요한 정보들만으로 잘 구성된 것으로 생각할 수 있다. 식 2.2는 의사결정 충실성을 나타낸다.

$$L_{F}(\tilde{X}, X, f) = log|f(\tilde{X}) - f(X)|$$
 (2.2)

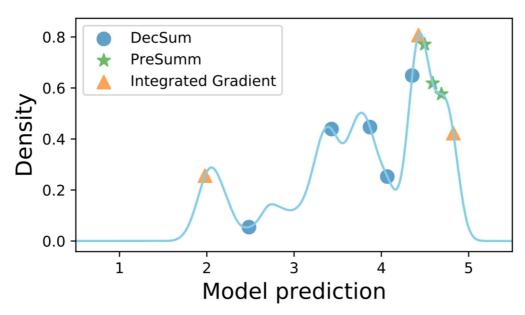


Figure 2.8: DecSum의 의사결정 대표성

의사결정 대표성은 그림 2.8과 같이 전체 텍스트로부터 얻어지는 의사결정의 다양성과 요약문으로부터 얻어지는 의사결정의 다양성이 유사해야 한다는 점이다. 레스토랑 리뷰 텍스트를 보면 전반적으로 긍정적일지라도 몇 문장들은 부정적인 경우들이 있을 수 있다. 의사결정 대표성은 요약문 안에 의사결정에 도움이 될다양한 측면에 문장들이 포함되었는지를 측정한다. 레스토랑 평점은 1, 2, 3, 4, 5점을 부여하게 되어 있고, 예측 모델에 전체 문장을 넣었을 때의 평점 분포와 요약문을 넣었을 때의 평점 분포 간 Wasserstein 거리[11]를 계산하여 분포 간차이를 식 2.3과 같이 정량화한다.

$$L_{R}(\widetilde{X}, X, f) = log(W(\widehat{Y}_{\widetilde{X}}, \widehat{Y}_{X}))$$
 (2.3)

의미적 유사성은 일반적인 문서 요약에서 중요하게 고려하는 요소로, 문서 전체로부터 얻어지는 정보들의 다양성과 요약문으로부터 얻어지는 정보들의 다양성이 유사해야 한다는 것이다. 의사결정 대표성과 유사해 보이지만, 의사결정과 관계없이 오로지 텍스트의 의미만 집중한다는 중요한 차이가 있다. 식 2.4는 DecSum에서 의미적 유사성을 측정한 방식을 나타낸다. s(x)는 SentenceBERT를 활용한 문장 표현이다.

$$L_{D}(\tilde{X}, X, f) = \sum_{\mathbf{x} \in \tilde{X}} \max_{\mathbf{x}' \in \tilde{X} - \{\mathbf{x}\}} \operatorname{cossim}(\mathbf{s}(\mathbf{x}), \mathbf{s}(\mathbf{x}'))$$
 (2.4)

위의 의사결정의 충실성과 대표성, 그리고 의미적 유사성을 정량화한 손실함수는 식 2.5와 같다.

$$L(\tilde{X}, X, f) = \alpha L_{F}(\tilde{X}, X, f) + \beta L_{R}(\tilde{X}, X, f) + \gamma L_{D}(\tilde{X}, X, f)$$
(2.5)

2.8 반도체 제조 환경에서의 자연어처리

시장의 고성능 반도체에 대한 수요가 증가함에 따라 기술적인 복잡도가 계속 증가하고 있다. 이에 따라 반도체 제조 환경에서 생산성과 품질 등 다양한 측면에서 발전이 요구되고 있다. 이에 따라 다양하고 귀중한 정보들을 잠재적으로 포함한 기술 문서가 많고, 전문가들은 이 기술 문서들을 이해하고 있어야 한다. 불행하게도 이런 문서들은 대부분 구조화되어 있지 않아, 소화하는데 쉽지 않다. 컴퓨터 과학, 특히 자연어처리 기술의 발전에 힘입어 비구조화된 문서들을 분석하여 전문가들을 도와줄 수 있는 지식 베이스를 자동으로 구축하는 것이 가능해졌다. Razouk와 Houssam의 연구[12]에서는 언어 모델과 그래프 임베딩을 활용하여 그림 2.9와 같은 반도체 산업에서 고장 원인 분석과 위험 평가를 위한 전문가 시스템을 제안한다.

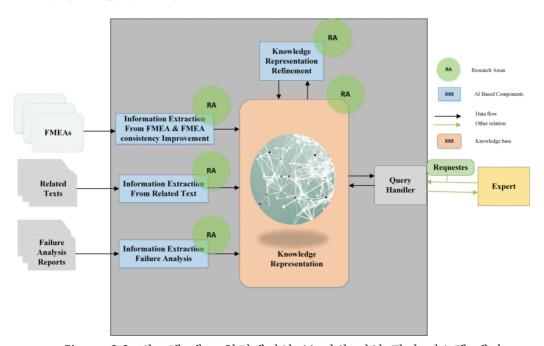


Figure 2.9: 반도체 제조 환경에서의 AI 기반 지식 관리 시스템 개념도

고장 분석은 여러 엔지니어의 발견한 현상과 결론에 대한 지식을 고려해야 하므로 굉장히 복잡하고 어렵다. Razouk와 Houssam의 연구[12]에 따르면, 다양한 분야의 전문가들이 기술 문서를 적기 때문에 고장에 의한 영향들의 순서가 바뀌거나, 다른 개념의 원인과 영향이 섞여 있는 등 문서의 일관성이 떨어지는 문제점들도 있다고 한다. 또한, 양식에 제한이 없어 이미지나 파일 공유, 위키 등다양한 정보를 담을 수 있어서 필요한 정보를 찾는데 쉽지 않다. Grabner와 Corinna의 연구[13]에서는 BERT를 미세 학습하여 고장 분석 문서를 전기적 상징과 물리적 실패의 종류를 분류하는 연구를 수행하였다.

Abu-Rasheed 등의 연구[14]에서는 반도체 칩 디자인과 관련된 문서로부터 지식을 추출하고, 체계적으로 관리하는 그래프 기반의 탐색을 제안하였다. 과거의 칩 디자인을 실패한 기록을 담은 문서를 분석하고 지식을 추출하여 현재 새로운 칩을 디자인하고 있는 사용자가 효율적으로 정보에 접근할 수 있도록 도움을 주는 목적이다. 지속적인 지식의 활용을 위해서는 단순히 비정형 문서로부터 지식을 뽑아내는 것뿐만 아니라 잠재적으로 알지 못하는 연결 관계까지 알아낼수 있어야 한다. 이를 위해 그림 2.10과 같은 칩 디자인 건과 프로젝트 건, 그리고 연결 노드들로 구성된 그래프를 활용하였다. 디자인 건 노드는 실패한 이전 칩 디자인으로부터 얻을 수 있는 지식을 대변한다. 이 노드는 어떤 실패였는지, 원인과 해결책과 같은 자세한 정보들을 포함한다. 매 실패한 경우마다 한 건씩 생겨난다. 프로젝트 노드는 디자인 프로젝트의 구조적인 의존 관계를 표현하는 메타 데이터이다. 프로젝트 노드는 프로젝트와 칩 모듈 간의 관계를 반영한다. 프로젝트 노드 간의 관계로부터 같은 프로젝트에 있든 다양한 프로젝트에 있든 라계없이 실패한 건들 간의 상관성을 알 수 있다. 연결 노드는 그래프 구조에

데이터마이닝 결과와 도메인 특화된 특징들을 넣는 요소이다. 연결 노드는 그래프상에서 과거와 미래의 디자인 건들 간 유사성을, 칩 디자인 문서에서 사용된 도메인 특화된 용어들을 표현한다. Al-Natsheh의 연구[15]와 같이 칩 디자인 간 관련도를 반영하고 유사성의 순위를 계산하기 위하여 N-gram과 기술 용어 분류기를 적용한 TFIDF를 활용하였다. 이를 통해 도메인 디자인 엔지니어의 작문 스타일과 기술적인 요소와 축약어 등 특화된 용어들과 문서 패턴들에 대응할 수 있도록 하였다.

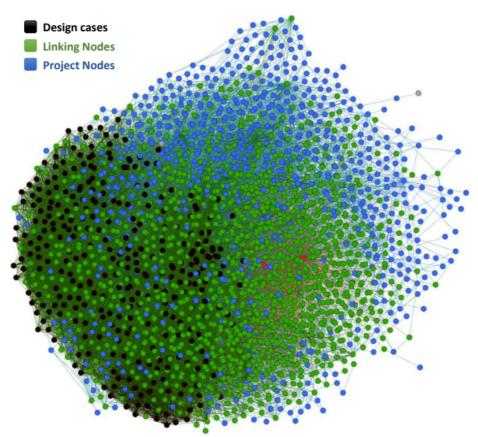


Figure 2.10: 칩 지식 그래프

제 3 장 제안하는 방법론

반도체 설비 운용 기록에서 조치 방법 중심의 요약문과 키워드를 추출하기 위하여 그림 3.1와 같이 전처리, 분류기, 최종 요약 및 키워드 추출 단계를 거친다 전처리 단계에서는 설비 운용 기록 안에 불필요한 정보들을 찾아 제거하고, 문장간 구분을 형태소 분석을 통해 보강한다. 분류기 단계에서는 문장이 조치 방법과관련되어 있는지 학습한다. 최종 요약 단계에서는 의미적 중복을 고려하여요약문에 포함할 문장들을 추출한다. 키워드 추출 단계에서는 설비 축약어와 설명데이터를 활용하여 키워드들을 추출, 효율적으로 검색될 수 있도록 한다.

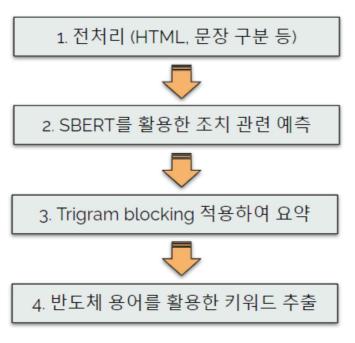


Figure 3.1: 제안하는 방법론

반도체 분야의 기술 문서를 다루는 것은 다양한 어려움에 직면한다. 특히 고려해야 하는 점은 바로 보안이다. 반도체 산업은 핵심 국가 산업이기에 더 그렇다. 모든 데이터는 사외로 보내기 위하여 변형이 필요하다. 텍스트는 변형하게 되면 제대로 된 분석을 하기 어려우므로, 연구 진행 방법에 대한 많은 고민이 있었다. 다행히도 사외에서 연구 환경을 갖춘 보안 서버에 접근할 수 있도록 지원이 되어 연구를 자유롭게 할 수 있었다.

앞서 관련 연구에서 서술한 것과 같이, 다양한 분야의 전문가들이 기술 문서를 적기 때문에 고장에 의한 영향들의 순서가 바뀌거나, 다른 개념의 원인과 영향이 섞여 있는 등 문서의 일관성이 떨어지는 문제점들도 있다. 또한, 양식에 제한이 없어 이미지나 파일 공유, 위키 등 다양한 정보를 담을 수 있어서 필요한 정보를 찾는데 쉽지 않다[13]. 실제 반도체 설비 운용 문서를 살펴보아도 같은 문제점을 발견할 수 있었다. 교대 근무를 하는 특성상 운용 문서를 작성하는 엔지니어가 계속 바뀌고, 운용 문서는 양식에 제한이 없어 자세히 자유롭게 적을 수 있는 장점이 있는 한편 후에 중요한 지식을 추출하는 등 시스템적으로 관리 되기가 쉽지 않은 단점이 있었다.

3.1 전처리

반도체 설비 운용 기록은 설비에 문제가 발생했을 때 문제의 원인, 현황 그리고 조치 방법 등의 있는 웹 문서이다. 일정한 규격이 없기에 그 내용도 자유로워 HTML 태그들, 설비 로그들, 시간 정보, 특수문자 등 조치 방법과 무관한 데이터들이 많이 포함되어 있다. 불필요한 데이터들을 제거하고 품질을 높이는 작업은 최종 요약 성능에 큰 영향을 주기 때문에 굉장히 중요하다.

Table 3.1: 전처리 과정

구분	내용
원문	<head>3/10 8시 13분 설비 Stack-1 에러가 발생, </head> <body>mask: 13242512.모듈 분해 후 확인 결과 CNM 모듈 이슈로 예상됨 9시 10분 Clamp h2 제거 후 점검. BPart에 스크래치 확인하여 Part 교체. 정상 동작 확인하여 이슈 close함. sensor value tilt 0.7 ##</body>
HTML 제거 후	3/10 8시 13분 설비 Stack-1 에러가 발생, mask: 13242512. 모듈 분해 후 확인 결과 CNM 모듈 이슈로 예상됨 9시 10분 Clamp h2 제거 후 점검. B Part에 스크래치 확인하여 Part 교체. 정상 동작 확인하여 이슈 close함. sensor value tilt 0.7 ##
설비 로그 등 영어/숫자만 있는 문장 제거 후	3/10 8시 13분 설비 Stack-1 에러가 발생, 모듈 분해 후 확인 결과 CNM 모듈 이슈로 예상됨 9시 10분 Clamp h2 제거 후 점검. B Part에 스크래치 확인하여 Part 교체. 정상 동작 확인하여 이슈 close함. sensor value tilt 0.7 ##
센서 데이터 테이블 제거 후	3/10 8시 13분 설비 Stack-1 에러가 발생, 모듈 분해 후 확인 결과 CNM 모듈 이슈로 예상됨 9시 10분 Clamp h2 제거 후 점검. B Part에 스크래치 확인하여 Part 교체. 정상 동작 확인하여 이슈 close함. ##
불필요한 특수문자 제거 후	3/10 8시 13분 설비 Stack-1 에러가 발생, 모듈 분해 후 확인 결과 CNM 모듈 이슈로 예상됨 9시 10분 Clamp h2 제거 후 점검. B Part에 스크래치 확인하여 Part 교체. 정상 동작 확인하여 이슈 close함.
형태소 분석, 문장 구분 후	3/10 8시 13분 설비 Stack-1 에러가 발생, 모듈 분해 후 확인 결과 CNM 모듈 이슈로 예상됨. 9시 10분 Clamp h2 제거 후 점검. B Part에 스크래치 확인하여 Part 교체. 정상 동작 확인하여 이슈 close함.

표 3.1은 전처리 과정을 나타낸 것이다. 먼저 정규표현식으로 HTML 태그들을 찾아 제거하였다. 설비 운용 기록은 모두 한국인이 작성한다는 전제가 있어, 영어나 숫자만 있는 문장과 특수문자들은 조치 방법과 무관하다고 판단하고 삭제하였다. 설비 로그에는 다양한 데이터들이 있는데, 예를 들어 '\'를 구분자로 하여 센서의 값 같은 것들이 있다. 운용 기록에 있는 이런 데이터들을 살펴본

결과 고장 조치 방식과 관련이 없다고 판단하여 설비 센서 데이터가 있는 데이블을 제거하였다. 마지막으로 마침표가 없어 문장 간 구분이 되어 있지 않은 경우가 많았는데, 문장의 끝이 '됨', '함'과 같은 경우들이 대부분이라 형태소 분석후 'ㅁ' 받침 후에 마침표를 생성하여 문장 간 구분이 될 수 있도록 하였다.

3.2 분류기

문장을 입력으로 넣었을 때 고장 조치와 관련된 정도를 예측하기 위하여 분류기를 학습하였다. 그림 3.2와 같이 일반적인 요약 방식은 문서의 전반적인 문맥을 골고루 담기 때문에 본 연구의 목적에 부합하지 않는다. 따라서 각 문장의고장 관련 정도를 예측하고, 높은 순서대로 요약문에 포함하는 방식을 선택하였다.

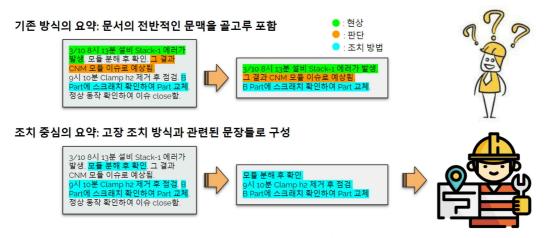


Figure 3.2: 조치 중심의 요약

학습은 설비 전문가가 직접 레이블링한 데이터를 활용하였다. 표 3.2는 학습 문장의 예시이다. 고장 조치와 관련된 문장의 수는 1451, 그렇지 않은 문장의 수는 2374개였다.

Table 3.2: 분류기 학습 데이터 예시

조치 관련 여부	418
0	3/10 8시 13분 설비 Stack-1 에러가 발생,
1	모듈 분해 후 확인.
0	결과 CNM 모듈 이슈로 예상됨.
1	9시 10분 Clamp h2 제거 후 점검.
1	B Part에 스크래치 확인하여 Part 교체.
0	정상 동작 확인하여 이슈 close함.

분류기는 SBERT를 선택하였다. 고장 조치와 관련 있거나 혹은 없는 문장들로 SBERT를 미세조정하였다. 성능을 실험한 분류기로는 SBERT로 임베딩한 벡터를 로지스틱 회귀한 것, SBERT, KoSBERT, KcELECTRA, KoSimCSE 였고 ROUGE 점수로 평가한 결과 SBERT의 성능이 가장 좋았다. 그 이유로 데이터가 한국어와 영어가 섞여 있는 다국어 문장이고 SBERT가 문장 임베딩에 특화된 점으로 판단하는데, 자세한 내용은 실험 결과에서 후술한다.

3.3 최종 요약 및 키워드 추출

분류기로 고장 조치 관련 정도를 예측하면 확률이 높은 순서대로, 의미적 중복을 고려하여 요약문에 포함할 문장들을 선택한다. 의미적 중복을 고려한다는 것은 요약문에 유사한 문장들이 없게끔 하는 것이다. Mihalcea와 Tarau의 연구[2], Carbonell과 Goldstein의 연구[8]에 착안하여 요약문에 선택할 문장을 탐욕적인 반복 알고리즘으로 선택하였다.

요약문에 포함할 문장의 수 k를 정하기 위하여 일반적으로 그리드 서치하지만 본 연구에서는 사용자의 요구사항에 따라 세 문장으로 정하였다. 더 적으면 중요한 정보가 포함되지 못할 염려가 있고, 더 많다면 불필요한 정보가 포함되거나 요약문을 읽을 때의 피로도를 고려한 것으로 보인다.

의미적 중복을 고려하기 위하여 문장 간 유사도를 측정해야 한다. 본연구에서는 표 3.4와 같이 트라이그램(Trigram) 블록킹을 사용하였다. 트라이그램 블록킹은 트라이그램이 중복되는 문장들은 유사하다고 판단하는 방식이다. 요약문에 이미 두 문장이 포함되어 있다고 하자. 조치 관련성이 가장 높은 후보문장을 요약문에 포함할 것인지 판단할 때, 이미 포함된 두 문장 간 트라이그램이 중복되는지 확인한다. 중복이라면 다음 조치 관련성이 높은 문장을 후보 문장으로 선택하고, 그렇지 않다면 요약문에 포함한다.

Table 3.3: 요약문에 포함된 문장 수에 따른 내용

문장 수	418
2	모듈 분해 후 확인. 9시 10분 Clamp h2 제거 후 점검.
3	모듈 분해 후 확인. 9시 10분 Clamp h2 제거 후 점검. B Part에 스크래치 확인하여 Part 교체.
4	모듈 분해 후 확인. 9시 10분 Clamp h2 제거 후 점검. B Part에 스크래치 확인하여 Part 교체. 결과 CNM 모듈 이슈로 예상됨.

Table 3.4: 트라이그램 블록킹의 예

순위	પાંક
1	모듈 분해 후 확인
2	B Part 스크래치 확인하여 Part 교체.
3	B Part 스크래치 이슈로 설비 초기화
4	설비 변수 값 초기화.
최종 요약	모듈 분해 후 확인. B Part 스크래치 확인하여 Part 교체. 설비 변수 값 초기화

설비 운용 기록들이 효율적으로 검색될 수 있도록 설비 축약어와 의미 데이터를 활용하여 키워드 추출을 하였다. 설비에 이슈가 발생했을 때 그 이슈에 대한 조치 방법들을 분류할 수 있다면 어떻게 대응할 것인지 의사결정에 도움이될 수 있다. 인스타그램의 해시태그는 같은 해시태그를 가진 사진들을 모아볼 수 있는 것처럼, 조치 방법 중심의 요약문으로부터 키워드들을 추출하게 되면 유사한 조치 방법끼리 분류하는 효과가 있다.

키워드 추출 방식은 칩 디자인 용어를 분류하는 분류기와 N-gram 접근 방법을 통해 도메인 특화된 지식을 그래프에 접목한 Abu-Rasheed 등의 연구[14]에 착안하여 표 3.5와 같은 설비의 축약어, 의미 데이터를 활용하였다.

그림 3.3과 같이 설비 축약어와 의미 데이터와 문장 간 겹치는 단어들을 키워드로 선정하는 방식이다. 이 방식을 채택한 이유는 일반적으로 설비의 하드웨어들은 축약어로 통용되고, 조치 방법을 나누는 중요한 기준 중 하나는

어떤 부품을 개선했는가이기 때문이다. 설비 운용 기록에는 설비에 대한 모델 정보도 있어서 모델에 따라 다른 하드웨어 특성들도 고려할 수 있는 장점이 있다.

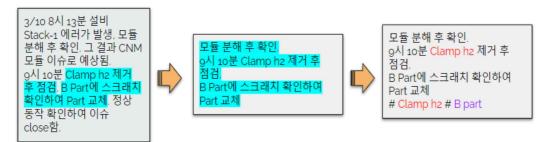


Figure 3.3: 설비 운용 기록으로부터의 키워드 추출 과정 예시

Table 3.5: 반도체 축약어와 의미 예시

축약어	의미
ALD	Atomic Layer Deposition
ALE	Atomic Layer Etch
AOI	Automated Optical Inspection
AP	Access Point
ASIC	Application Specific Integrated Circuit
ATE	Automatic Test Equipment
BEOL	Back-End-Of-Line
BGA	Ball Grid Array
ВР	B Part
BSA	Basic Service Area
BTI	Bias-Temperature Instability

제 4 장 실험 및 결과

본 장에서는 설비 운용 기록을 앞서 제안한 방법으로 요약한 결과를 정답 요약문과 비교하여 평가한다. 먼저 정량적으로 활용하여 텍스트랭크, BERTSUM와 같은 일반적인 요약 방식과 ROUGE 점수를 측정한다. 그리고 설비 운용 기록과 정답 요약문, 시스템 요약문들을 비교 분석하여 본 연구의 우수성을 보인다.

4.1 실험 세팅 및 평가 지표

먼저 분류기를 학습하기 위해 조치 방법과 관련 있는 문장 1451건, 관련 없는 문장 2374건을 활용하였다. 학습은 각 레이블별로 2/3씩 데이터를 임의로 선택하였고, 남은 데이터로 테스트를 진행하였다. 분류기 학습 시 배치 크기를 8, AdamW 옵티마이저, 가중치 감쇠는 0.01, 학습률(Learning Rate)은 0.001, 500 스텝의 warmup 선형 스케줄러를 사용하였다. SBERT로 임베딩을 하고 로지스틱 회귀를 적용한 모델, SBERT, KoSBERT, KcELECTRA, KoSimCSE를 각각 미세학습한 분류기들을 실험하였다.

ROUGE(Recall-Oriented Understudy for Gisting Evaluation) 점수는 시스템 요약문과 정답 요약문 간 얼마나 유사한지 평가하는 지표이다[16]. 유사성을 판단할 때 단어 간 겹침을 기준으로 하는데, 그렇게 된다면 시스템 요약문이 다양한 단어들을 많이 담을수록 값이 올라가기 때문에 정밀도와 재현율, F1 점수를 함께 고려한다(식 4.1). 0에서 1의 값을 가지며 높을수록 요약이 잘 되어 있는 것이다. 단어의 기준은 유니그램(Unigram), 바이그램(Bigram), 최장 공통

부문 문자열(Longest Common Subsequences; LCS) 세 가지 방법으로 평가하였다. 유니그램을 사용한 경우 ROUGE-1, 바이그램을 사용한 경우 ROUGE-2, 최장 공통 부문 문자열을 사용한 경우 ROUGE-L로 표현하였다.

4.2 평가 결과 및 요약 예시

정답 요약문이 있는 653건의 설비 운용 기록들을 활용하여 실험을 진행하였다. 그 결과 SBERT를 분류기로 활용한 요약 모델의 ROUGE-1, ROUGE-2, ROUGE-L 모두 가장 좋았다. 또, 일반적인 요약 모델의 성능보다 고장 조치 중심의 요약 모델들의 성능이 더 높음을 알 수 있었다. 고장 조치 중심의 요약 모델 중 SBERT가 가장 좋은 결과를 보인 것은 SBERT가 문장 단위의 임베딩에 특화되어 있고 요약문이 한글과 영어가 혼용되었기 때문으로 예상한다.

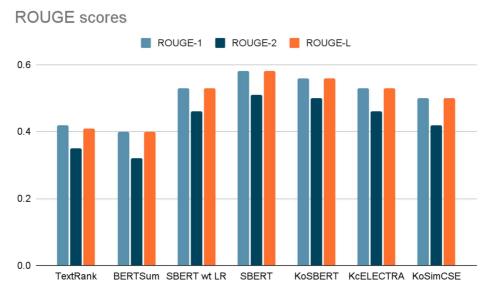


Figure 4.1: 시스템 요약문 평가 결과

Table 4.1: 시스템 요약문 평가 결과

A lawa with ma	ROUGE-1			ROUGE-2			ROUGE-L		
Algorithm	R	Р	F1	R	Р	F1	R	Р	F1
TextRank	0.51	0.50	0.42	0.44	0.44	0.35	0.51	0.49	0.41
BERTSum	0.40	0.52	0.40	0.33	0.43	0.32	0.40	0.52	0.40
SBERT with LR	0.61	0.62	0.53	0.54	0.56	0.46	0.61	0.61	0.53
SBERT	0.67	0.66	0.58	0.61	0.60	0.51	0.67	0.67	0.58
KoSBERT	0.61	0.66	0.56	0.56	0.60	0.50	0.61	0.66	0.56
KcELECTRA	0.61	0.61	0.53	0.54	0.56	0.46	0.10	0.61	0.53
KoSimCSE	0.58	0.57	0.50	0.51	0.50	0.42	0.58	0.57	0.50

표 4.2와 4.3는 요약 예시이다. 먼저 요약문 전 문서를 통해 어떻게 고장에 대하여 조치하였는지 파악하기는 쉽지 않은 일임을 이해할 수 있다. 반면에 정답 요약문에는 조치 방식들만이 요약되어 있어 한눈에 어떤 조치를 하였는지 알 수 있다. 문서의 문맥을 고루 반영하는 일반적인 요약 모델인 BERTSUM의 요약문을 정답 요약문과 비교하면, 중요한 조치 방법이 누락 되어 있고 불필요한 정보들이

포함되어 있음을 확인할 수 있다. 본 논문에서 제안하는 Action-Focused Summarization의 요약문을 살펴보면 정답 요약문과 굉장히 유사함을 알 수 있다. 그렇지만 표 4.2.2의 'Hub screen 면 오염에 의한 Hub Weight 차이.'와 같이 불필요한 문장을 포함하는 한계가 있다. 이는 요약문에 포함할 문장의 수를 일괄적으로 세 문장으로 하였기 때문이고, 이를 고장 조치와 관련성을 고려하여 유동적으로 보완할 수 있을 것으로 생각한다.

Table 4.2: 요약문 예시

구분	418
원문	Stack-1 Error발생 등 Hub fix시 Stack-1 Error가 발생함. Hub fix시 계측된 Hub MS Point의 Position이 Expected Position 대비 차이가 Threshold를 넘어감. 해당 Threshold는 설정에서 변경이 가능하나 통상적으로 10nm~20nm 정도임. 가능한 원인은 1. Hub screen 면 오염에 의한 Hub Weight 차이 2. Hub Side Tilt 등에 의한 Position Drift가 있음. Hub screen면 오염인 경우 특정 Hub에서 Trend가 상승하며, Side 의 Vacuum screen의 오염인 경우 전반적인 Trend가 상승한다. Hub Side, Tama Side 등 설비 구조적인 문제에서 유발되는 경우에도 전반적인 Trend가 상승한다. 본건은 전반적인 Trend가 상승한 경우로, HS Vacuum screen의 오염이나 TS, HS 등의 구조적인 문제로 의심할 수 있다. 1차적으로 Vacuum screen Clean을 실시하고, 개선이 없을 경우 DESO와 DESO에 따른 Recovery로 Focal 작업을 필요로 한다.
정답 요약	1차적으로 Vacuum screen Clean을 실시하고, 개선이 없을 경우 DESO와 DESO에 따른 Recovery로 Focal 작업을 필요로 한다.
BERTSUM 요약	Stack-1 Error발생 등 Hub fix시 Stack-1 Error가 발생함. Hub fix시 계측된 Hub MS Point의 Position이 Expected Position 대비 차이가 Threshold를 넘어감. Hub screen면 오염인 경우 특정 Hub에서 Trend가 상승하며, Side 의 Vacuum screen의 오염인 경우 전반적인 Trend가 상승한다.
본 연구의 요약	1차적으로 Vacuum screen Clean을 실시하고, 개선이 없을 경우 DESO와 DESO에 따른 Recovery로 Focal 작업을 필요로 한다. Hub screen 면 오염에 의한 Hub Weight 차이.

Table 4.3: 요약문 예시

구분	내용
원문	MS side h2 drop에 의한 fixment fial. Side fix시 MS에 h2 drop이 있다고 판단되는 경우 이를 제거하기 위해 MS drop decay 작업을 진행. 1. 발생 원인 - MS side h2 drop에 의한 fixment fail - Robot 오염 - 2. 대책작업 Case 1 : 특정 MS side 에서만 발생하는 경우 - MS side cleaning. Case 2: MS side 구분 없이 발생 - Robot sony cleaning - Robot/MS side 오염으로 인해서 MS side fixment Point의 위치에 h2가 남겨짐. h2 drop 상태에서 MS fixment/Tilt를 계측하게 되면 low Tilt jump 됨. 0.7 이하일 경우 MS side를 RS side으로 이동하여 air shower를 이용하여 Dry.
정답 요약	대책작업 Case 1 : 특정 MS side 에서만 발생하는 경우 - MS side cleaning. Case 2: MS side 구분 없이 발생 - Robot sony cleaning - Robot/MS side 오염으로 인해서 MS side fixment Point의 위치에 h2가 남겨짐. h2 drop 상태에서 MS fixment/Tilt를 계측하게 되면 low Tilt jump 됨
BERTSUM 요약	MS side h2 drop에 의한 fixment fial. Side fix시 MS에 h2 drop이 있다고 판단되는 경우 이를 제거하기 위해 MS drop decay 작업을 진행. 발생 원인 - MS side h2 drop에 의한 fixment fail - Robot 오염. h2 drop 상태에서 MS fixment/Tilt를 계측하게 되면 low Tilt jump 됨.
본 연구의 요약	대책작업 Case 1 : 특정 MS side 에서만 발생하는 경우 - MS side cleaning. Case 2: MS side 구분 없이 발생. 0.7 이하일 경우 MS side를 RS side으로 이동하여 air shower를 이용하여 Dry. MS side h2 drop에 의한 fixment fail. Side fix시 MS에 h2 drop이 있다고 판단되는 경우 이를 제거하기 위해 MS drop decay 작업을 진행.

설비 축약어와 설명 데이터를 활용하여 요약문으로부터 키워드를 추출하는 과정은 정답 데이터 셋이 없어 정량적인 평가가 이뤄지지 못했다. 하지만 설비 전문가들에게 아이디어를 설명하고 그림 4.2와 같은 추출 결과를 보여준 결과 굉장히 긍정적으로 반응하였다. 전문가 의견에 따르면 키워드가 설비의 하드웨어 요소에 대응되는 효과가 있고 원문이 아닌 요약문으로부터 키워드 추출을 수행하여 조치 방법에 따른 군집화 효과도 있을 것으로 예상한다고 한다. 다만설비 용어가 설비의 발전에 따라 지속해서 변경이 됨에 따라, 사용자가 직접

키워드에 대응되는 용어들을 유지 보수할 방법이 고안되어야 할 것으로 보인다.

모듈 분해 후 확인. 9시 10분 Clamp h2 제거 후 점검. B Part에 스크래치 확인하여 Part 교체 # Clamp h2 # B part

Load robot 교체. Clamp h2 제거 후 재초기화. 테스트 결과 이상 없음을 확인. # Load robot # Clamp h2

Figure 4.2: 키워드 추출 결과 예시

제 5 장 결론

반도체 설비 고장이 발생하면 엔지니어는 고장과 관련된 모듈 전문가의 의견이나, 설비 운용 기록을 참고할 수 있다. 회로 패턴이 미세해짐에 따라 하드웨어 구조가 복잡해지고 상대적으로 전문가의 수는 부족하다. 따라서 비정형 데이터인 설비 운용 기록들을 분석하고 분류하여 유용한 지식의 형태로 유지 및 관리될 수 있도록 하는 것은 굉장히 중요하다.

본 연구에서는 설비 운용 기록 중 가장 중요한 내용인 고장 조치 방법을 중심으로 요약하고, 효율적인 검색을 위해 키워드를 추출하는 프레임워크를 제안한다. 일반적인 요약 방식은 문서의 전반적인 내용을 고루 담아 요약하는 것을 목표로 하므로, 연구 목표에 적합하지 않음을 확인하였다. 따라서 문장을 임베딩 벡터로 만드는 SBERT를 미세 학습하여 문장의 고장 조치 여부를 예측할 수 있도록 하였다. 학습된 분류기를 통해 고장 조치 관련성을 예측하고, 요약문의 포함될 세 문장을 관련성이 높은 순서대로 선택하였다. 요약문의 문장 수를 세문장으로 정한 이유는 필요한 정보가 빠지거나 불필요한 정보가 포함될 가능성을 고려한 사용자의 요구사항 중 하나였다. 또한, 요약문 간 유사한 문장이 포함될 수 있기에, 트라이그램 블록킹을 적용하여 이를 방지할 수 있도록 하였다. 시스템 요약문의 성능을 ROUGE 점수를 기반으로 평가한 결과 일반적인 요약 방식의결과가 비교하여 약 0.2 정도 개선된 것을 확인하였다. 그리고 정성적으로 요약문을 살펴보았을 때도, Action-Focused Summarization을 적용한 결과가 정답 요약문과 유사하며 이를 통해서 조치 방식을 손쉽게 이해할 수 있음을 확인하였다. 마지막으로 효과적인 검색 및 분류를 위해, 최종 요약문으로부터 설비

축약어와 의미 정보를 활용하여 키워드를 추출할 수 있도록 하였다.

본 연구의 의의는 다음과 같다. 먼저, 설비 운용 기록들을 시스템적으로 분석하고 요약 및 키워드를 추출하는 최초의 시도이다. 설비 운용 기록들은 설비에 대한 지식이 녹아 있는 아카이브이기 때문에, 이들을 효과적으로 검색될수 있고 좋은 품질의 형태로 유지될 수 있도록 하는 것은 중요하다. 두 번째로, 설비 운용 기록을 고장 조치 방법을 중심으로 요약하여 손쉽게 기록을 이해하고 파악할 수 있게 하였다. 설비 운용 기록은 최소 월 삼만 건이 생성되고, 양식에 제한이 없어 다양한 내용이 포함될 수 있으므로 설비 엔지니어가 가장 중요하게 참고하는 고장 조치 방식을 이해하기가 쉽지 않다. 시스템을 활용해 자동으로 고장 조치 중심의 요약을 하여 제공된다면 설비 고장에 더 빨리 대응할 수 있다. 세 번째로, 조치 방법과 관련 있는 하드웨어를 중심으로 키워드를 추출하여 효과적으로 필요한 정보가 검색될 수 있도록 하였다. 추출된 키워드들을 인스타그램의 해시태그처럼 활용한다면, 유사한 조치를 한 기록끼리 군집화하는 효과도 있게 된다.

본 연구의 한계점은 다음과 같다. 먼저, 요약문의 문장의 수를 세 문장으로 고정해두었기 때문에, 기록마다 불필요한 정보가 포함되거나 필요한 정보가 빠지는 경우가 있다. 이를 요약문의 포함되는 문장의 수를 예측되는 문장별 고장조치 관련성을 고려, 유동적으로 설정하여 개선할 수 있을 것으로 생각한다. 또한, 설비 운용 기록 중 이미지에 포함된 조치 방식들은 빠진다. 기록들을 살펴본 결과간혹 프레젠테이션에 적혀 있는 조치 방식들을 그대로 캡처하여 이미지로 첨부한경우들이 존재하였다. 이를 요약 전에 포함된 이미지들을 광학 문자 인식(Optical character recognition; OCR)으로 텍스트화하여 개선이 가능할 것이다.

참고 문헌

- [1] Ran, Yongyi, et al. "A survey of predictive maintenance: Systems, purposes and approaches." arXiv preprint arXiv:1912.07383 (2019).
- [2] Mihalcea, Rada, and Paul Tarau. "Textrank: Bringing order into text."

 Proceedings of the 2004 conference on empirical methods in natural language processing. 2004.
- [3] Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).
- [4] Liu, Yang. "Fine-tune BERT for extractive summarization." arXiv preprint arXiv:1903.10318 (2019).
- [5] ReimeHS, Nils, and Iryna Gurevych. "Sentence-bert: Sentence embeddings using siamese bert-networks." arXiv preprint arXiv:1908.2084 (2019).
- [6] Clark, Kevin, et al. "Electra: Pre-training text encoders as discriminators rather than generators." arXiv preprint arXiv:2003.10555 (2020).
- [7] Gao, Tianyu, Xingcheng Yao, and Danqi Chen. "SimCSE: Simple contrastive learning of sentence embeddings." arXiv preprint arXiv:2104.08821 (2021).
- [8] Jaime Carbonell and Jade Goldstein. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In Proceedings of the 21st annual international ACM SIGIR conference on

- Research and development in information retrieval, pages 335-336.
- [9] Dina Demner-Fushman, Wendy W Chapman, and Clement J McDonald. 2009. What can natural language processing do for clinical decision support? Journal of biomedical informatics, 42(5):760-772.
- [10] Hsu, Chao-Chun, and Chenhao Tan. "Decision-Focused Summarization." arXiv preprint arXiv:2109.06896 (2021).
- [11] Aaditya Ramdas, Nicolás Trillos, and Marco Cuturi. 2017. On wasserstein two-sample testing and related families of nonparametric tests. Entropy, 19(2):4
- [12] Razouk, Houssam, et al. "Al-Based Knowledge Management System for Risk Assessment and Root Cause Analysis in Semiconductor Industry."

 Artificial Intelligence for Digitising Industry-Applications. River Publishers, 2022. 113-129.
- [13] Grabner, Corinna, et al. "A BERT-Based Report Classification for Semiconductor Failure Analysis." ISTFA 2022. ASM International, 2022.
- [14] Abu-Rasheed, Hasan, et al. "Explainable Graph-Based Search for Lessons-Learned Documents in the Semiconductor Industry." Intelligent Computing. Springer, Cham, 2022. 1097-1106.
- [15] H. Al-Natsheh, "Text Mining Approaches for Semantic Similarity Exploration and Metadata Enrichment of Scientific Digital Libraries," p. 176, 2019

[16] Lin, Chin-Yew. "Rouge: A package for automatic evaluation of summaries." Text summarization branches out. 2004.

Abstract

SBERT based action focused summarization and keyword extraction for knowledge management in semiconductor facility

Nuri Han
Department of Industrial Engineering
The Graduate School
Seoul National University

Among facility operation records, the information on handling failure is important to be referred to similar failures. Since thousands of facility operation records are produced monthly in semiconductor production environment, it is practically impossible to analyze action methods without system. Also, general summarization method can only capture an overall context of the document. In this study, we proposed a framework that learns SBERT to classify whether it is related to action, summarizes documents focused on action methods, and extracts keywords from the summary using semiconductor terminology information. We obtained improved ROUGE scores than general summarization techniques. As a result of this study, improvement of productivity is expected by easily acquiring necessary information and by enabling stable knowledge management.

Keywords: Extractive Summarization, Keyword Extraction, Preprocessing

Student Number: 2021-21051