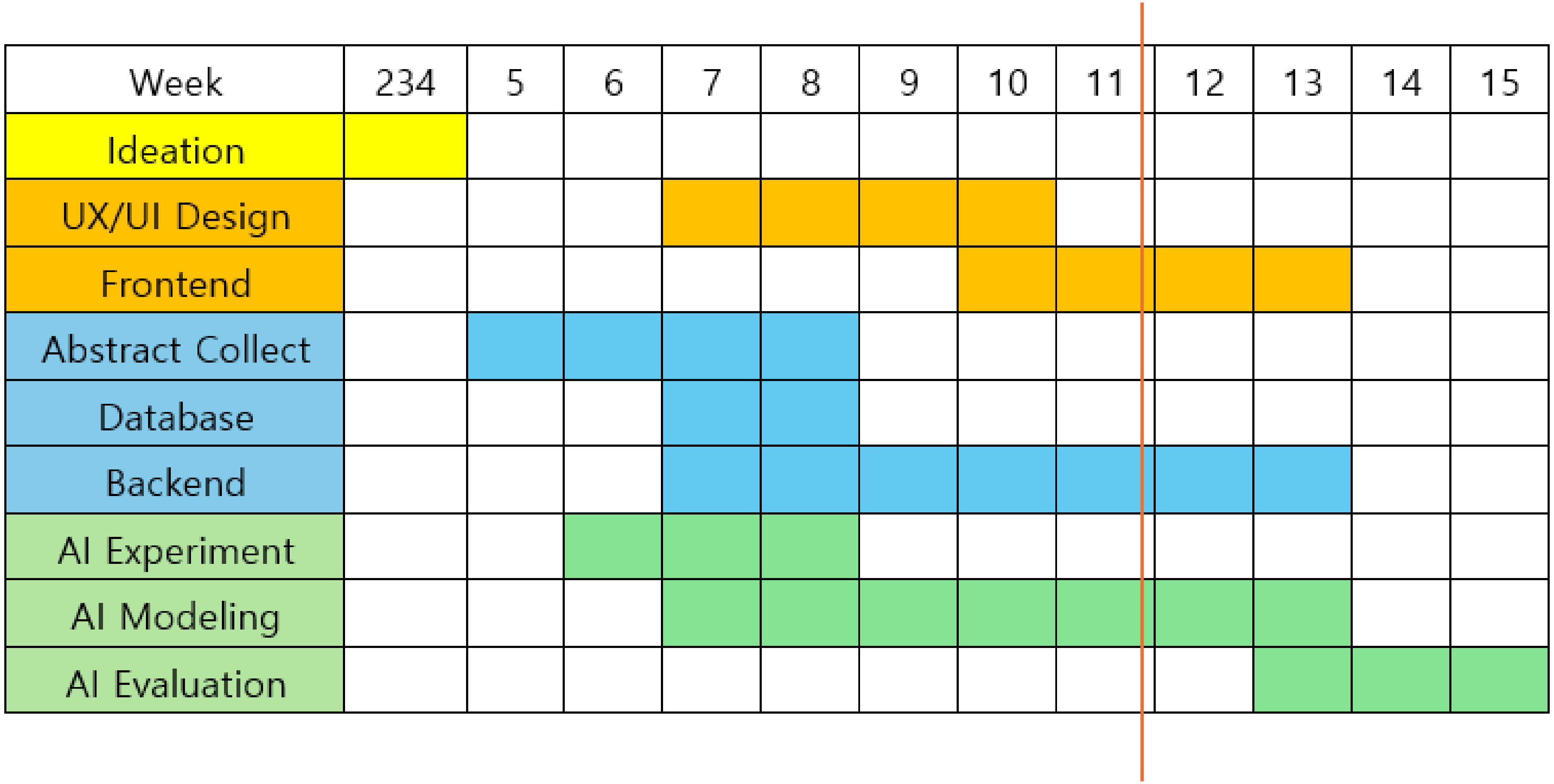
SKKU Lab Recommendation Service : FindMyLab

Team J (Last Lap) 김현진 송민석 장민석 조재희

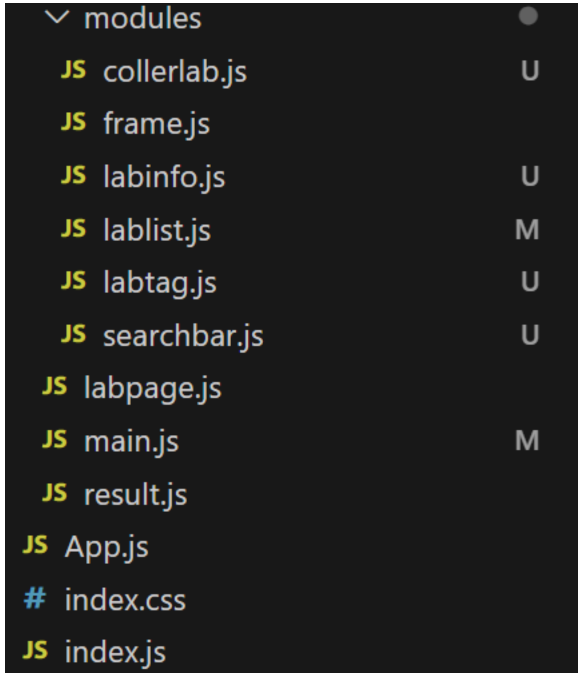
2024-Fall Capstone Design Project

1. Objectives & Milestones

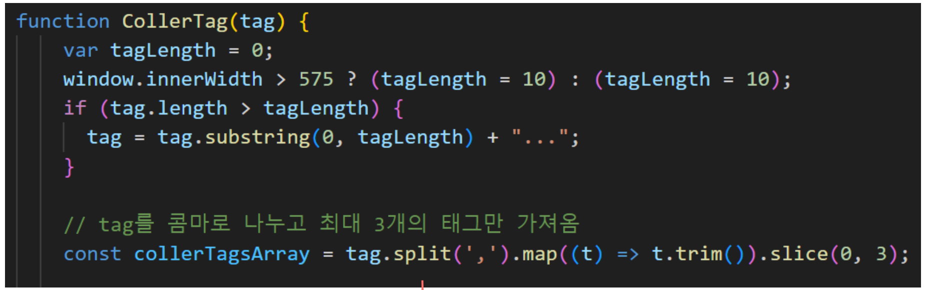


Week	234	5	6	7	8	9	10	11	12	13	14	15
Ideation	🟨											
UX/UI Design				🟧	🟧	🟧	🟧					
Frontend							🟧	🟧	🟧	🟧		
Abstract Collect		🟦	🟦	🟦	🟦							
Database				🟦	🟦							
Backend				🟦	🟦	🟦	🟦	🟦	🟦	🟦		
AI Experiment			🟩	🟩	🟩							
AI Modeling				🟩	🟩	🟩	🟩	🟩	🟩	🟩		
AI Evaluation										🟩	🟩	🟩

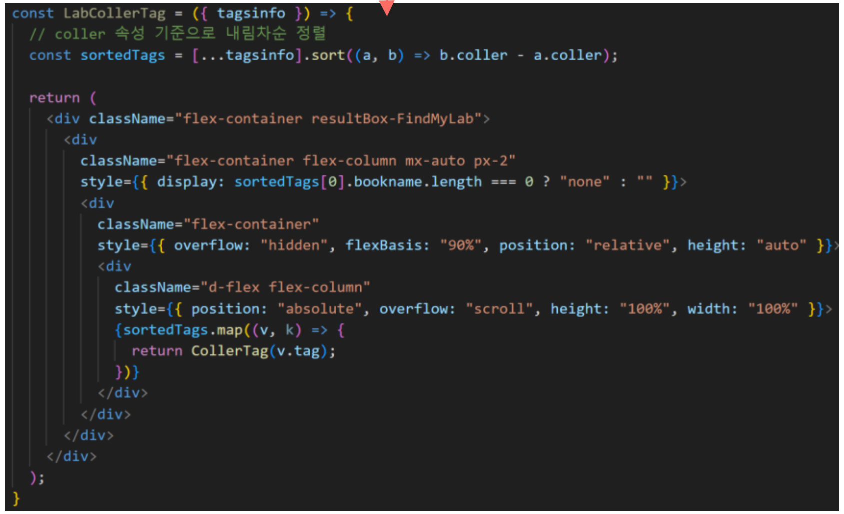
2. Frontend / Component-Based Abstraction



```
∨ modules                ●
    JS collerlab.js      U
    JS frame.js
    JS labinfo.js        U
    JS lablist.js        M
    JS labtag.js         U
    JS searchbar.js      U
  JS labpage.js
  JS main.js             M
  JS result.js
JS App.js
# index.css
JS index.js
```

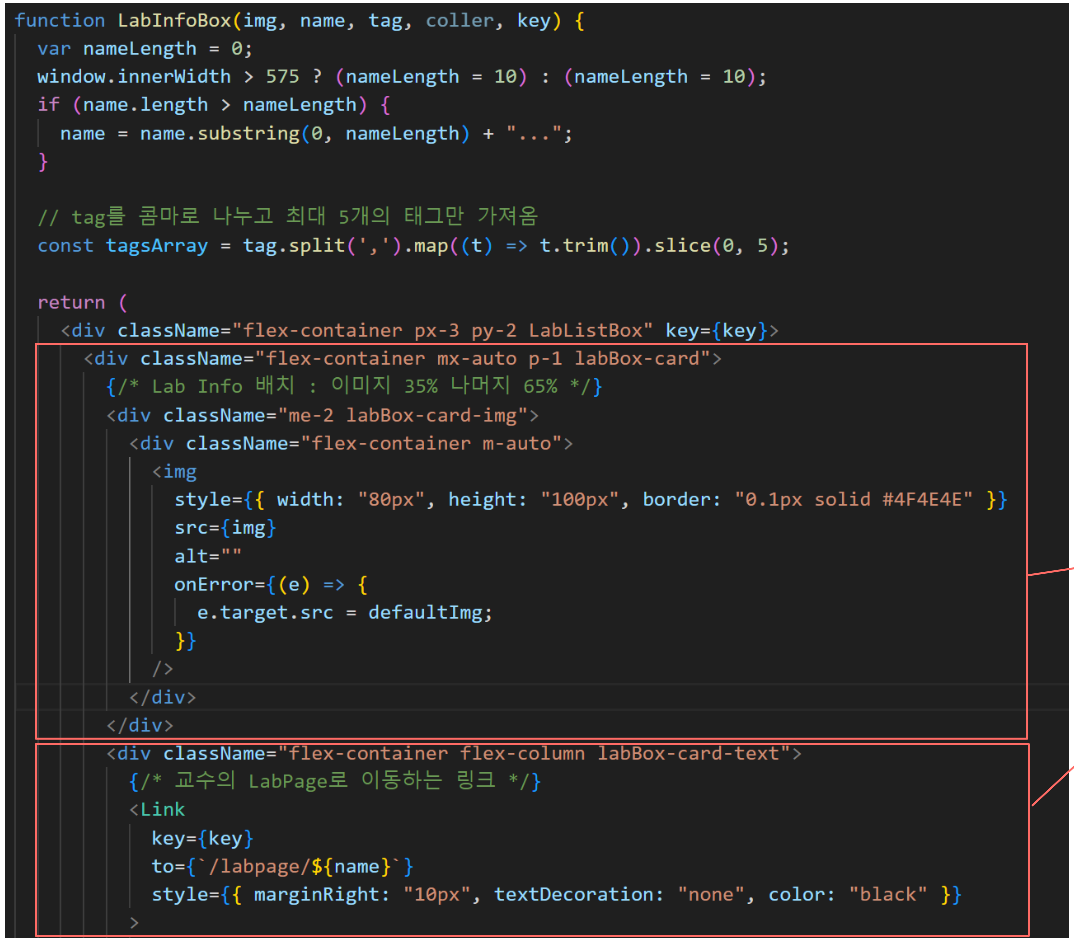


```javascript
function CollerTag(tag) {
    var tagLength = 0;
    window.innerWidth > 575 ? (tagLength = 10) : (tagLength = 10);
    if (tag.length > tagLength) {
      tag = tag.substring(0, tagLength) + "...";
    }

    // tag를 콤마로 나누고 최대 3개의 태그만 가져옴
    const collerTagsArray = tag.split(',').map((t) => t.trim()).slice(0, 3);
```

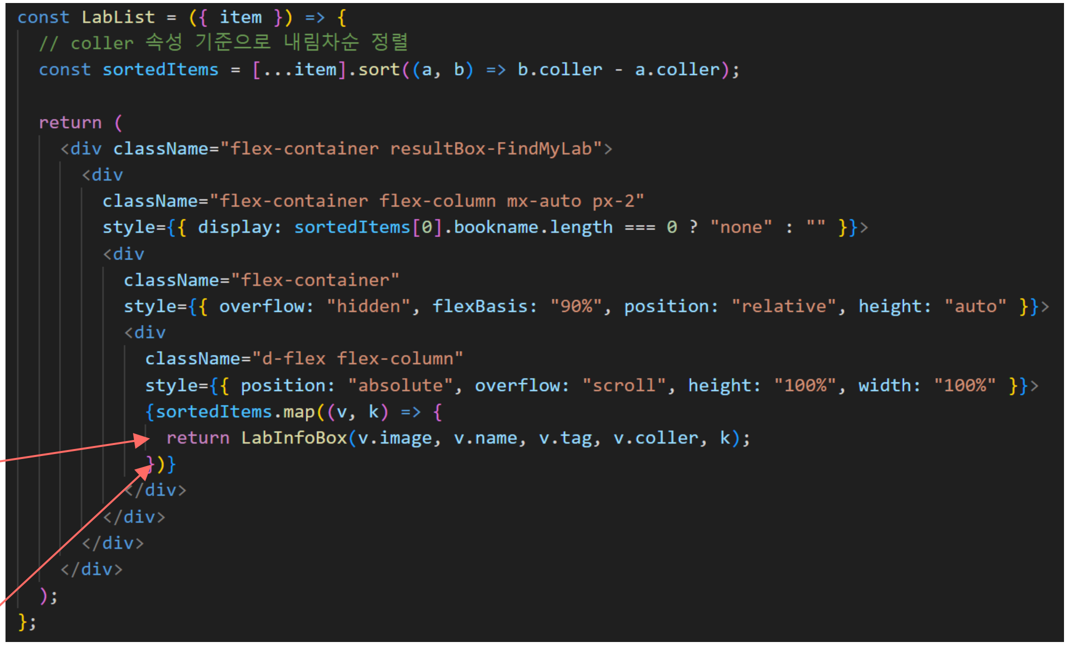


```javascript
const LabCollerTag = ({ tagsinfo }) => {
  // coller 속성 기준으로 내림차순 정렬
  const sortedTags = [...tagsinfo].sort((a, b) => b.coller - a.coller);

  return (
    <div className="flex-container resultBox-FindMyLab">
      <div
        className="flex-container flex-column mx-auto px-2"
        style={{ display: sortedTags[0].bookname.length === 0 ? "none" : "" }}>
        <div
          className="flex-container"
          style={{ overflow: "hidden", flexBasis: "90%", position: "relative", height: "auto" }}>
          <div
            className="d-flex flex-column"
            style={{ position: "absolute", overflow: "scroll", height: "100%", width: "100%" }}>
            {sortedTags.map((v, k) => {
              return CollerTag(v.tag);
            })}
          </div>
        </div>
      </div>
    </div>
  );
}
```

2. Frontend / Parameterization

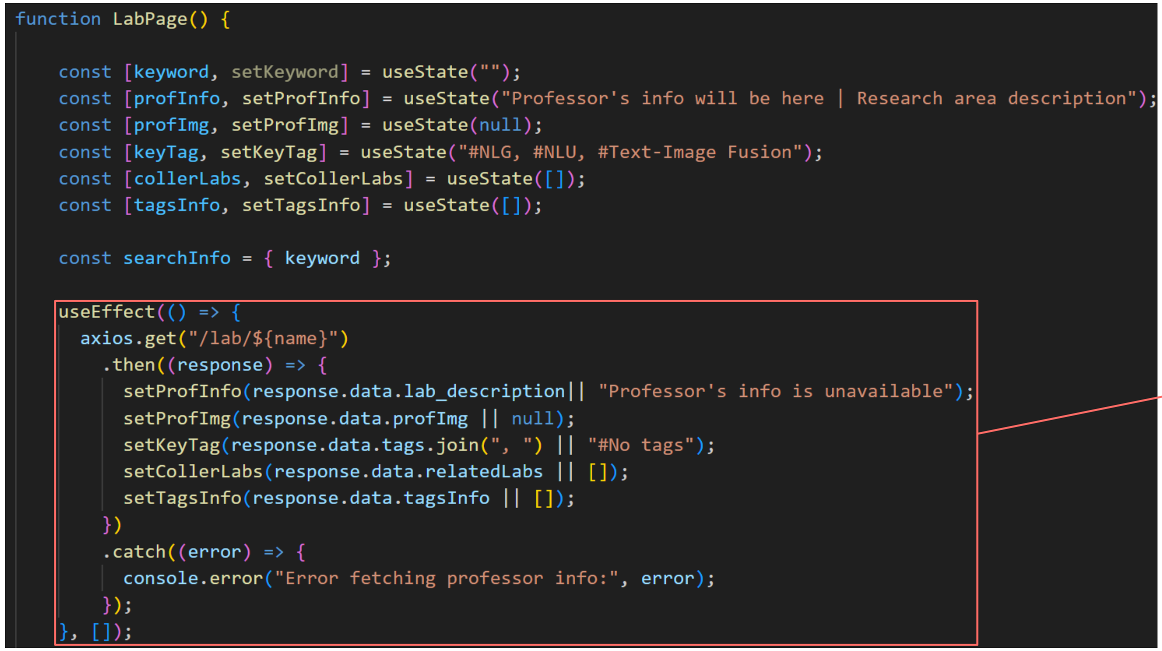


```
function LabInfoBox(img, name, tag, coller, key) {
  var nameLength = 0;
  window.innerWidth > 575 ? (nameLength = 10) : (nameLength = 10);
  if (name.length > nameLength) {
    name = name.substring(0, nameLength) + "...";
  }

  // tag를 콤마로 나누고 최대 5개의 태그만 가져옴
  const tagsArray = tag.split(',').map((t) => t.trim()).slice(0, 5);

  return (
    <div className="flex-container px-3 py-2 LabListBox" key={key}>
      <div className="flex-container mx-auto p-1 labBox-card">
        {/* Lab Info 배치 : 이미지 35% 나머지 65% */}
        <div className="me-2 labBox-card-img">
          <div className="flex-container m-auto">
            <img
              style={{ width: "80px", height: "100px", border: "0.1px solid #4F4E4E" }}
              src={img}
              alt=""
              onError={(e) => {
                e.target.src = defaultImg;
              }}
            />
          </div>
        </div>
        <div className="flex-container flex-column labBox-card-text">
          {/* 교수의 LabPage로 이동하는 링크 */}
          <Link
            key={key}
            to={`/labpage/${name}`}
            style={{ marginRight: "10px", textDecoration: "none", color: "black" }}
          >
```

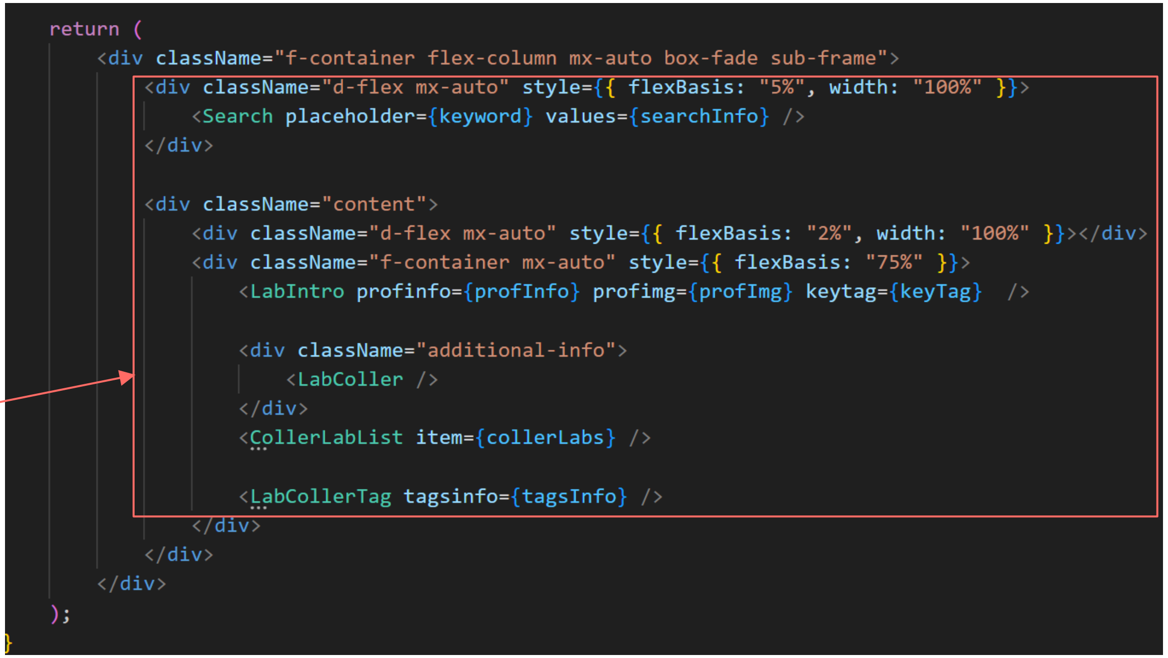


```
const LabList = ({ item }) => {
  // coller 속성 기준으로 내림차순 정렬
  const sortedItems = [...item].sort((a, b) => b.coller - a.coller);

  return (
    <div className="flex-container resultBox-FindMyLab">
      <div
        className="flex-container flex-column mx-auto px-2"
        style={{ display: sortedItems[0].bookname.length === 0 ? "none" : "" }}>
        <div
          className="flex-container"
          style={{ overflow: "hidden", flexBasis: "90%", position: "relative", height: "auto" }}>
          <div
            className="d-flex flex-column"
            style={{ position: "absolute", overflow: "scroll", height: "100%", width: "100%" }}>
            {sortedItems.map((v, k) => {
              return LabInfoBox(v.image, v.name, v.tag, v.coller, k);
            })}
          </div>
        </div>
      </div>
    </div>
  );
};
```

2. Frontend / State-Driven UI

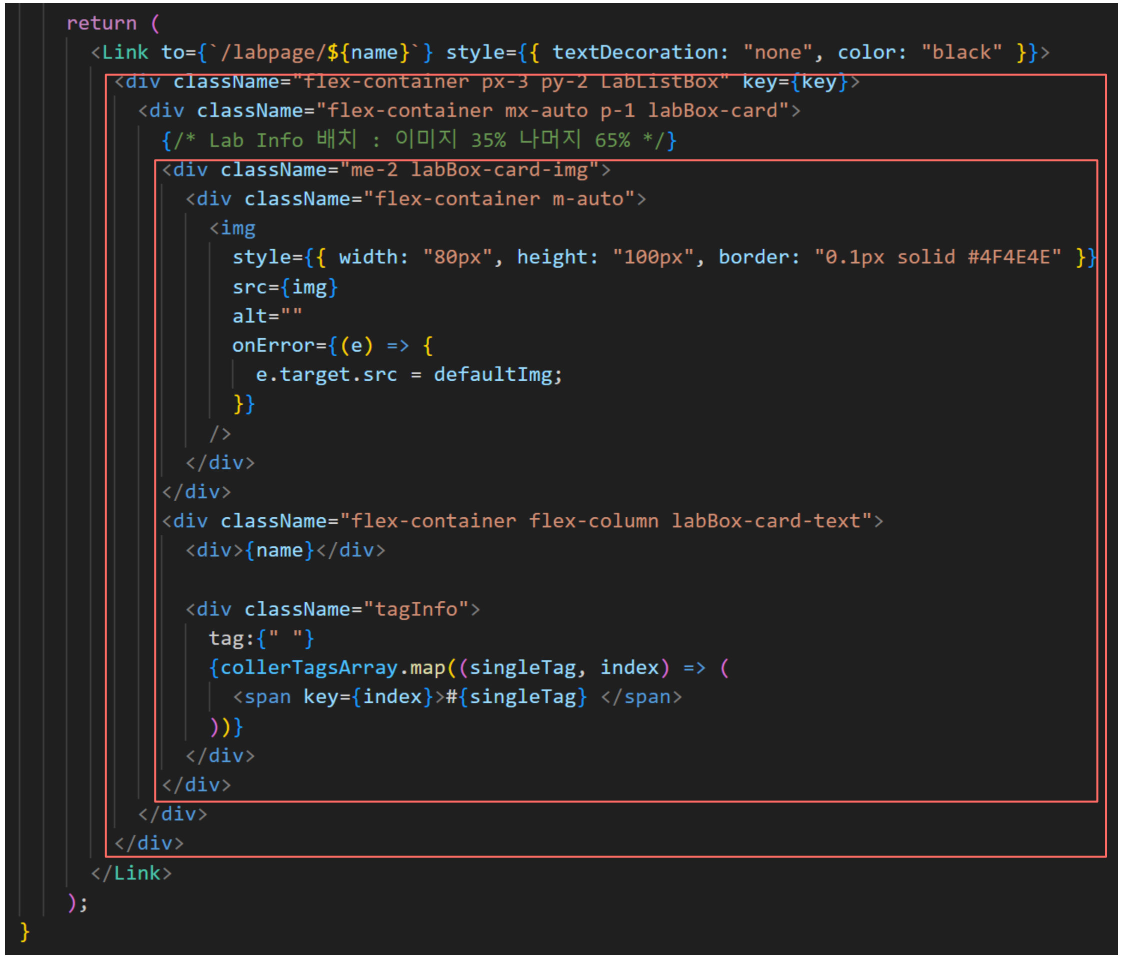


```
function LabPage() {

    const [keyword, setKeyword] = useState("");
    const [profInfo, setProfInfo] = useState("Professor's info will be here | Research area description");
    const [profImg, setProfImg] = useState(null);
    const [keyTag, setKeyTag] = useState("#NLG, #NLU, #Text-Image Fusion");
    const [collerLabs, setCollerLabs] = useState([]);
    const [tagsInfo, setTagsInfo] = useState([]);

    const searchInfo = { keyword };

    useEffect(() => {
      axios.get("/lab/${name}")
        .then((response) => {
          setProfInfo(response.data.lab_description|| "Professor's info is unavailable");
          setProfImg(response.data.profImg || null);
          setKeyTag(response.data.tags.join(", ") || "#No tags");
          setCollerLabs(response.data.relatedLabs || []);
          setTagsInfo(response.data.tagsInfo || []);
        })
        .catch((error) => {
          console.error("Error fetching professor info:", error);
        });
    }, []);
```



```
    return (
        <div className="f-container flex-column mx-auto box-fade sub-frame">
            <div className="d-flex mx-auto" style={{ flexBasis: "5%", width: "100%" }}>
                <Search placeholder={keyword} values={searchInfo} />
            </div>

            <div className="content">
                <div className="d-flex mx-auto" style={{ flexBasis: "2%", width: "100%" }}></div>
                <div className="f-container mx-auto" style={{ flexBasis: "75%" }}>
                    <LabIntro profinfo={profInfo} profimg={profImg} keytag={keyTag}  />

                    <div className="additional-info">
                        <LabColler />
                    </div>
                    <CollerLabList item={collerLabs} />

                    <LabCollerTag tagsinfo={tagsInfo} />
                </div>
            </div>
        </div>
    );
}
```

2. Frontend / Hierarchical Structure

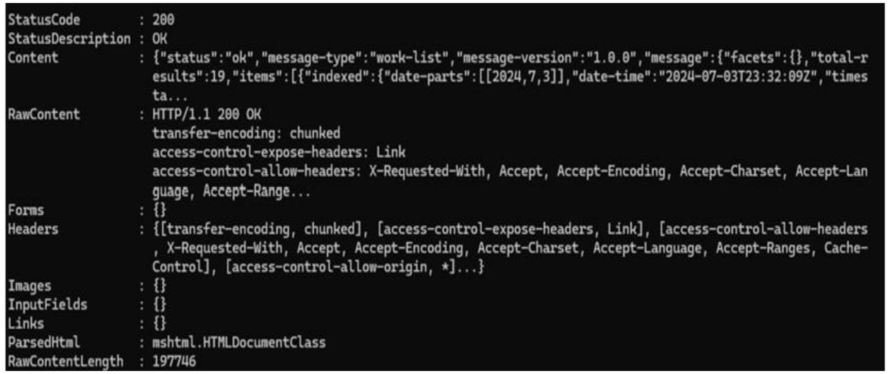


```
    return (
      <Link to={`/labpage/${name}`} style={{ textDecoration: "none", color: "black" }}>
        <div className="flex-container px-3 py-2 LabListBox" key={key}>
          <div className="flex-container mx-auto p-1 labBox-card">
            {/* Lab Info 배치 : 이미지 35% 나머지 65% */}
            <div className="me-2 labBox-card-img">
              <div className="flex-container m-auto">
                <img
                  style={{ width: "80px", height: "100px", border: "0.1px solid #4F4E4E" }}
                  src={img}
                  alt=""
                  onError={(e) => {
                    e.target.src = defaultImg;
                  }}
                />
              </div>
            </div>
            <div className="flex-container flex-column labBox-card-text">
              <div>{name}</div>

              <div className="tagInfo">
                tag:{" "}
                {collerTagsArray.map((singleTag, index) => (
                  <span key={index}>#{singleTag} </span>
                ))}
              </div>
            </div>
          </div>
        </div>
      </Link>
    );
}
```

3. Backend / Abstract Extract - previous algorithm

ORCID

Connecting Research
and Researchers

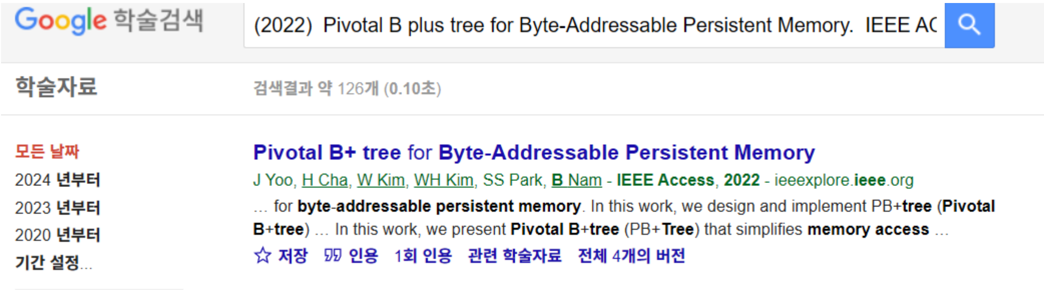


```
StatusCode        : 200
StatusDescription : OK
Content           : {"status":"ok","message-type":"work-list","message-version":"1.0.0","message":{"facets":{},"total-r
                    esults":19,"items":[{"indexed":{"date-parts":[[2024,7,3]],"date-time":"2024-07-03T23:32:09Z","times
                    ta...
RawContent        : HTTP/1.1 200 OK
                    transfer-encoding: chunked
                    access-control-expose-headers: Link
                    access-control-allow-headers: X-Requested-With, Accept, Accept-Encoding, Accept-Charset, Accept-Lan
                    guage, Accept-Range...
Forms             : {}
Headers           : {[transfer-encoding, chunked], [access-control-expose-headers, Link], [access-control-allow-headers
                    , X-Requested-With, Accept, Accept-Encoding, Accept-Charset, Accept-Language, Accept-Ranges, Cache-
                    Control], [access-control-allow-origin, *]...}
Images            : {}
InputFields       : {}
Links             : {}
ParsedHtml        : mshtml.HTMLDocumentClass
RawContentLength  : 197746
```

[Crossref API]

학술지 논문

- (2022) Pivotal B plus tree for Byte-Addressable Persistent Memory. IEEE ACCESS. 10, -
- (2021) Failure-Atomic Byte-Addressable R-tree for Persistent Memory. IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS. 32, 3
- (2021) Failure-Atomic Byte-Addressable R-tree for Persistent Memory. IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS. 32, 3
- (2020) B-3-Tree: Byte-Addressable Binary B-Tree for Persistent Memory. ACM TRANSACTIONS ON STORAGE. 16, 3



(2022) Pivotal B plus tree for Byte-Addressable Persistent Memory. IEEE AC

학술자료

검색결과 약 126개 (0.10초)

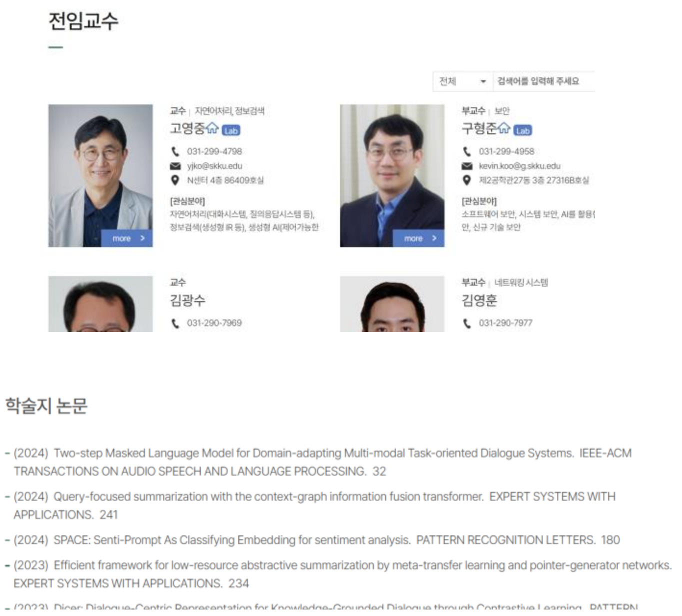
모든 날짜
2024 년부터
2023 년부터
2020 년부터
기간 설정...

Pivotal B+ tree for Byte-Addressable Persistent Memory

J Yoo, H Cha, W Kim, WH Kim, SS Park, B Nam - **IEEE Access**, **2022** - ieeexplore.**ieee**.org

... for **byte-addressable persistent memory**. In this work, we design and implement PB+**tree** (**Pivotal B**+**tree**) ... In this work, we present **Pivotal B**+**tree** (PB+**Tree**) that simplifies **memory access** ...

3. Backend / Abstract Extract - SKKU Web Crawling



전임교수

전체 ▾ | 검색어를 입력해 주세요

교수 | 자연어처리, 정보검색
고영중 Lab
☎ 031-299-4798
✉ yjko@skku.edu
N센터 4층 86409호실
[관심분야]
자연어처리(대화시스템, 질의응답시스템 등), 정보검색(생성형 IR 등), 생성형 AI(제어가능한

more >

부교수 | 보안
구형준 Lab
☎ 031-299-4958
✉ kevin.koo@g.skku.edu
제2공학관27동 3층 27316B호실
[관심분야]
소프트웨어 보안, 시스템 보안, AI를 활용한 보안, 신규 기술 보안

more >

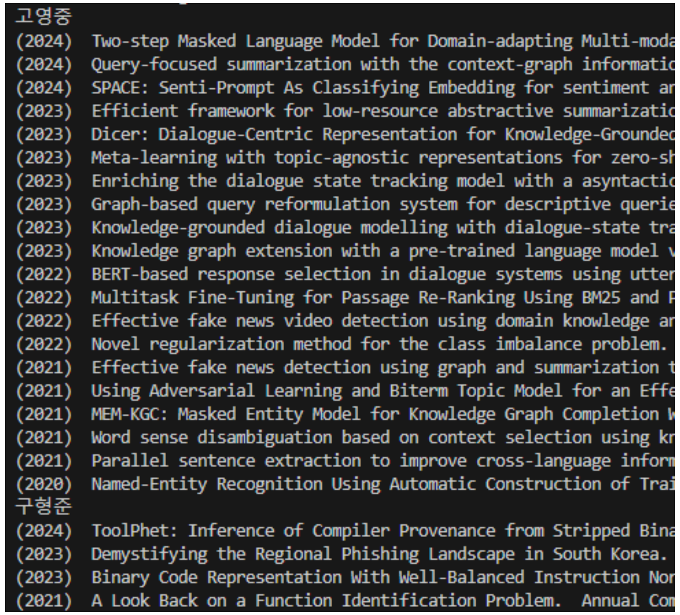
교수
김광수
☎ 031-290-7969

부교수 | 네트워킹 시스템
김영훈
☎ 031-290-7977

학술지 논문

- (2024) Two-step Masked Language Model for Domain-adapting Multi-modal Task-oriented Dialogue Systems. IEEE-ACM TRANSACTIONS ON AUDIO SPEECH AND LANGUAGE PROCESSING. 32
- (2024) Query-focused summarization with the context-graph information fusion transformer. EXPERT SYSTEMS WITH APPLICATIONS. 241
- (2024) SPACE: Senti-Prompt As Classifying Embedding for sentiment analysis. PATTERN RECOGNITION LETTERS. 180
- (2023) Efficient framework for low-resource abstractive summarization by meta-transfer learning and pointer-generator networks. EXPERT SYSTEMS WITH APPLICATIONS. 234
- (2023) Dicer: Dialogue-Centric Representation for Knowledge-Grounded Dialogue through Contrastive Learning. PATTERN

[SKKU Computer Science Web Site]

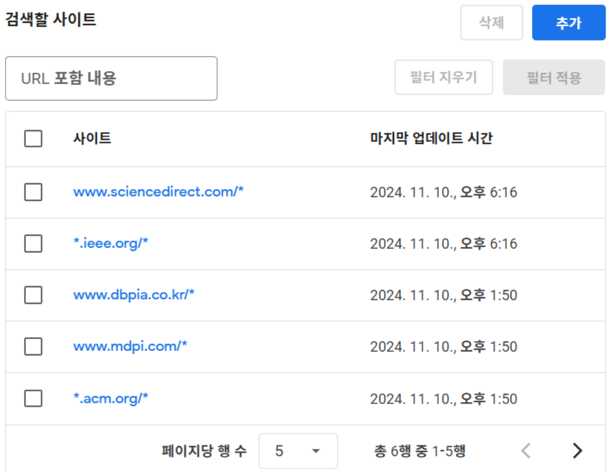


```
고영중
(2024)  Two-step Masked Language Model for Domain-adapting Multi-moda
(2024)  Query-focused summarization with the context-graph informatio
(2024)  SPACE: Senti-Prompt As Classifying Embedding for sentiment ar
(2023)  Efficient framework for low-resource abstractive summarizatio
(2023)  Dicer: Dialogue-Centric Representation for Knowledge-Grounded
(2023)  Meta-learning with topic-agnostic representations for zero-sh
(2023)  Enriching the dialogue state tracking model with a asyntactic
(2023)  Graph-based query reformulation system for descriptive querie
(2023)  Knowledge-grounded dialogue modelling with dialogue-state tra
(2023)  Knowledge graph extension with a pre-trained language model v
(2022)  BERT-based response selection in dialogue systems using utter
(2022)  Multitask Fine-Tuning for Passage Re-Ranking Using BM25 and P
(2022)  Effective fake news video detection using domain knowledge ar
(2022)  Novel regularization method for the class imbalance problem.
(2021)  Effective fake news detection using graph and summarization t
(2021)  Using Adversarial Learning and Biterm Topic Model for an Effe
(2021)  MEM-KGC: Masked Entity Model for Knowledge Graph Completion W
(2021)  Word sense disambiguation based on context selection using kn
(2021)  Parallel sentence extraction to improve cross-language inform
(2020)  Named-Entity Recognition Using Automatic Construction of Trai
구형준
(2024)  ToolPhet: Inference of Compiler Provenance from Stripped Bina
(2023)  Demystifying the Regional Phishing Landscape in South Korea.
(2023)  Binary Code Representation With Well-Balanced Instruction Nor
(2021)  A Look Back on a Function Identification Problem.  Annual Com
```

[Crawling Data]

3. Backend / Abstract Extract - Custom Search API

Custom Search API

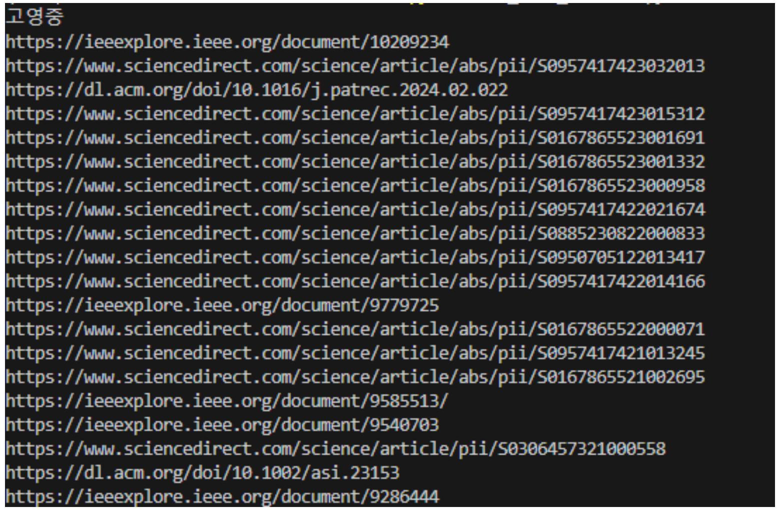


검색할 사이트

☐	사이트	마지막 업데이트 시간
☐	www.sciencedirect.com/*	2024. 11. 10., 오후 6:16
☐	*.ieee.org/*	2024. 11. 10., 오후 6:16
☐	www.dbpia.co.kr/*	2024. 11. 10., 오후 1:50
☐	www.mdpi.com/*	2024. 11. 10., 오후 1:50
☐	*.acm.org/*	2024. 11. 10., 오후 1:50

페이지당 행 수 5 총 6행 중 1-5행

[Custom Search Engine]



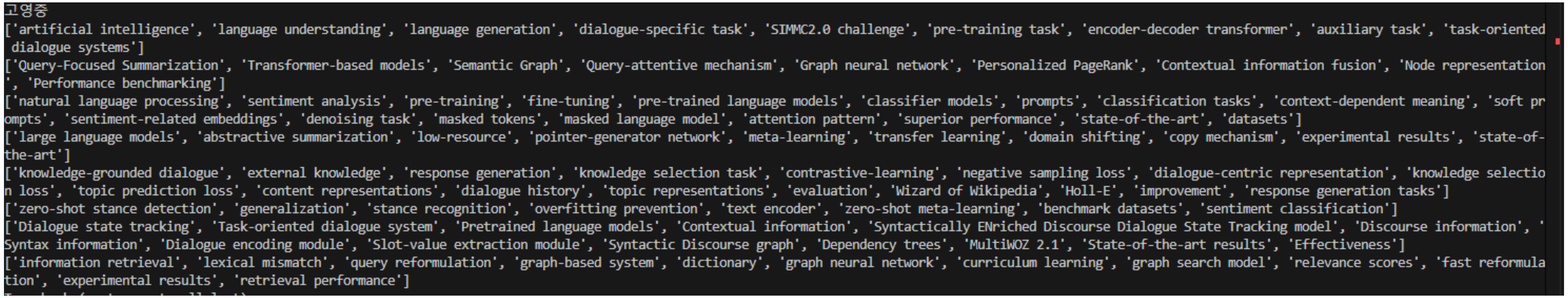
```
고영중
https://ieeexplore.ieee.org/document/10209234
https://www.sciencedirect.com/science/article/abs/pii/S0957417423032013
https://dl.acm.org/doi/10.1016/j.patrec.2024.02.022
https://www.sciencedirect.com/science/article/abs/pii/S0957417423015312
https://www.sciencedirect.com/science/article/abs/pii/S0167865523001691
https://www.sciencedirect.com/science/article/abs/pii/S0167865523001332
https://www.sciencedirect.com/science/article/abs/pii/S0167865523000958
https://www.sciencedirect.com/science/article/abs/pii/S0957417422021674
https://www.sciencedirect.com/science/article/abs/pii/S0885230822000833
https://www.sciencedirect.com/science/article/abs/pii/S0950705122013417
https://www.sciencedirect.com/science/article/abs/pii/S0957417422014166
https://ieeexplore.ieee.org/document/9779725
https://www.sciencedirect.com/science/article/abs/pii/S0167865522000071
https://www.sciencedirect.com/science/article/abs/pii/S0957417421013245
https://www.sciencedirect.com/science/article/abs/pii/S0167865521002695
https://ieeexplore.ieee.org/document/9585513/
https://ieeexplore.ieee.org/document/9540703
https://www.sciencedirect.com/science/article/pii/S0306457321000558
https://dl.acm.org/doi/10.1002/asi.23153
https://ieeexplore.ieee.org/document/9286444
```

[URI Data]

3. Backend - Abstract Extract - GPT prompting

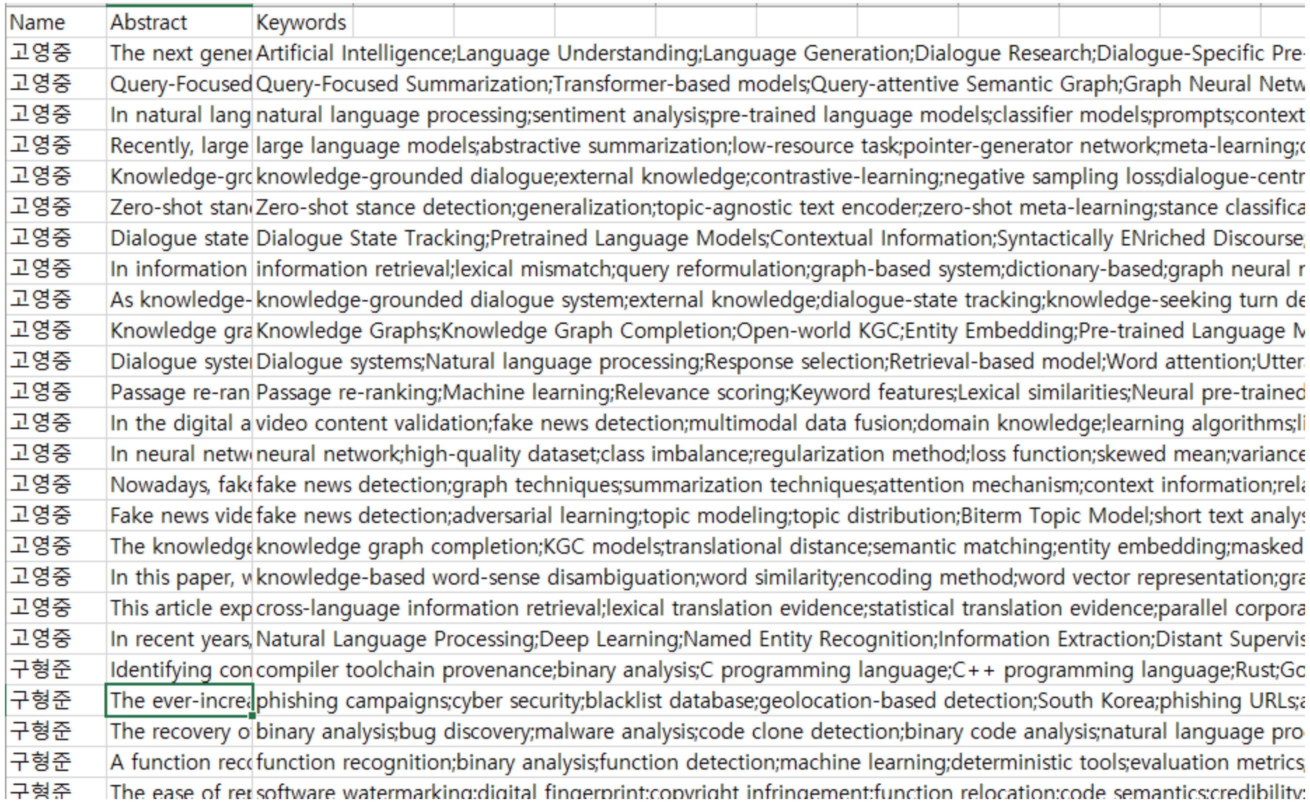
Prompt

Read the abstract of the provided research paper and return the {field} core technology and classification in a list format. Provide only concise keywords or phrases for each response item without full sentences.Read the abstract of the provided research paper and return the core technology and classification as a single, combined list of keywords in array format, without any headings or categorization



```
고영중
['artificial intelligence', 'language understanding', 'language generation', 'dialogue-specific task', 'SIMMC2.0 challenge', 'pre-training task', 'encoder-decoder transformer', 'auxiliary task', 'task-oriented
 dialogue systems']
['Query-Focused Summarization', 'Transformer-based models', 'Semantic Graph', 'Query-attentive mechanism', 'Graph neural network', 'Personalized PageRank', 'Contextual information fusion', 'Node representation
', 'Performance benchmarking']
['natural language processing', 'sentiment analysis', 'pre-training', 'fine-tuning', 'pre-trained language models', 'classifier models', 'prompts', 'classification tasks', 'context-dependent meaning', 'soft pr
ompts', 'sentiment-related embeddings', 'denoising task', 'masked tokens', 'masked language model', 'attention pattern', 'superior performance', 'state-of-the-art', 'datasets']
['large language models', 'abstractive summarization', 'low-resource', 'pointer-generator network', 'meta-learning', 'transfer learning', 'domain shifting', 'copy mechanism', 'experimental results', 'state-of-
the-art']
['knowledge-grounded dialogue', 'external knowledge', 'response generation', 'knowledge selection task', 'contrastive-learning', 'negative sampling loss', 'dialogue-centric representation', 'knowledge selectio
n loss', 'topic prediction loss', 'content representations', 'dialogue history', 'topic representations', 'evaluation', 'Wizard of Wikipedia', 'Holl-E', 'improvement', 'response generation tasks']
['zero-shot stance detection', 'generalization', 'stance recognition', 'overfitting prevention', 'text encoder', 'zero-shot meta-learning', 'benchmark datasets', 'sentiment classification']
['Dialogue state tracking', 'Task-oriented dialogue system', 'Pretrained language models', 'Contextual information', 'Syntactically ENriched Discourse Dialogue State Tracking model', 'Discourse information', '
Syntax information', 'Dialogue encoding module', 'Slot-value extraction module', 'Syntactic Discourse graph', 'Dependency trees', 'MultiWOZ 2.1', 'State-of-the-art results', 'Effectiveness']
['information retrieval', 'lexical mismatch', 'query reformulation', 'graph-based system', 'dictionary', 'graph neural network', 'curriculum learning', 'graph search model', 'relevance scores', 'fast reformula
tion', 'experimental results', 'retrieval performance']
```

3. Backend - Abstract Extract - GPT prompting

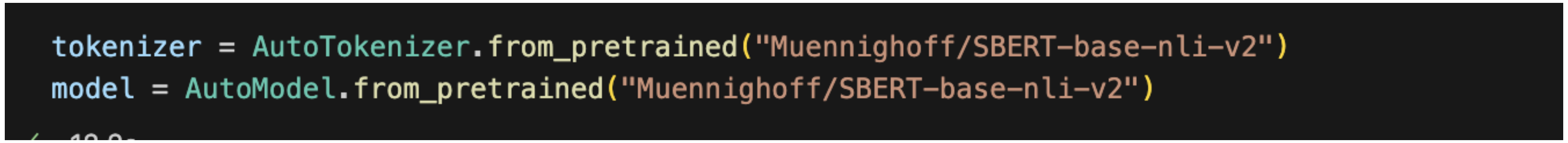


Name	Abstract	Keywords
고영중	The next gener	Artificial Intelligence;Language Understanding;Language Generation;Dialogue Research;Dialogue-Specific Pre-
고영중	Query-Focused	Query-Focused Summarization;Transformer-based models;Query-attentive Semantic Graph;Graph Neural Netw
고영중	In natural lang	natural language processing;sentiment analysis;pre-trained language models;classifier models;prompts;context
고영중	Recently, large	large language models;abstractive summarization;low-resource task;pointer-generator network;meta-learning;c
고영중	Knowledge-grc	knowledge-grounded dialogue;external knowledge;contrastive-learning;negative sampling loss;dialogue-centr
고영중	Zero-shot stan	Zero-shot stance detection;generalization;topic-agnostic text encoder;zero-shot meta-learning;stance classifica
고영중	Dialogue state	Dialogue State Tracking;Pretrained Language Models;Contextual Information;Syntactically ENriched Discourse;
고영중	In information	information retrieval;lexical mismatch;query reformulation;graph-based system;dictionary-based;graph neural r
고영중	As knowledge-	knowledge-grounded dialogue system;external knowledge;dialogue-state tracking;knowledge-seeking turn de
고영중	Knowledge gra	Knowledge Graphs;Knowledge Graph Completion;Open-world KGC;Entity Embedding;Pre-trained Language M
고영중	Dialogue syste	Dialogue systems;Natural language processing;Response selection;Retrieval-based model;Word attention;Utter
고영중	Passage re-ran	Passage re-ranking;Machine learning;Relevance scoring;Keyword features;Lexical similarities;Neural pre-trained
고영중	In the digital a	video content validation;fake news detection;multimodal data fusion;domain knowledge;learning algorithms;li
고영중	In neural netw	neural network;high-quality dataset;class imbalance;regularization method;loss function;skewed mean;variance
고영중	Nowadays, fak	fake news detection;graph techniques;summarization techniques;attention mechanism;context information;rela
고영중	Fake news vide	fake news detection;adversarial learning;topic modeling;topic distribution;Biterm Topic Model;short text analys
고영중	The knowledge	knowledge graph completion;KGC models;translational distance;semantic matching;entity embedding;masked
고영중	In this paper, w	knowledge-based word-sense disambiguation;word similarity;encoding method;word vector representation;gra
고영중	This article exp	cross-language information retrieval;lexical translation evidence;statistical translation evidence;parallel corpora
고영중	In recent years,	Natural Language Processing;Deep Learning;Named Entity Recognition;Information Extraction;Distant Supervis
구형준	Identifying con	compiler toolchain provenance;binary analysis;C programming language;C++ programming language;Rust;Go
구형준	The ever-increa	phishing campaigns;cyber security;blacklist database;geolocation-based detection;South Korea;phishing URLs;a
구형준	The recovery o	binary analysis;bug discovery;malware analysis;code clone detection;binary code analysis;natural language pro
구형준	A function reco	function recognition;binary analysis;function detection;machine learning;deterministic tools;evaluation metrics;
구형준	The ease of rei	software watermarking;digital fingerprint;copyright infringement;function relocation;code semantics;credibility;

4. AI / Search Algorithm

Model

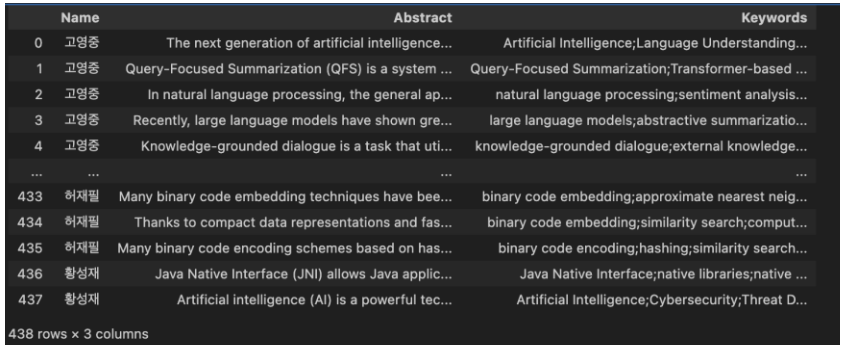
SBERT - nil pretrained model



```
tokenizer = AutoTokenizer.from_pretrained("Muennighoff/SBERT-base-nli-v2")
model = AutoModel.from_pretrained("Muennighoff/SBERT-base-nli-v2")
```

Data

NAME	Abstract	Keywords
고영중	The next generation..	Artificial intelligence, Language ..
고영중	Query-Focused Summarization is a system…	Transformer-based..
..		
…		
황성재	Java Native Interface allows Java…	Native libraries, Java Native interface



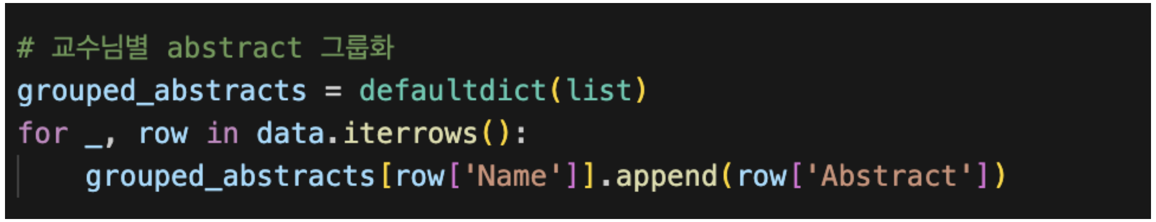
	Name	Abstract	Keywords
0	고영중	The next generation of artificial intelligence...	Artificial Intelligence;Language Understanding...
1	고영중	Query-Focused Summarization (QFS) is a system ...	Query-Focused Summarization;Transformer-based ...
2	고영중	In natural language processing, the general ap...	natural language processing;sentiment analysis...
3	고영중	Recently, large language models have shown gre...	large language models;abstractive summarizatio...
4	고영중	Knowledge-grounded dialogue is a task that uti...	knowledge-grounded dialogue;external knowledge...
...	...	...	...
433	허재필	Many binary code embedding techniques have bee...	binary code embedding;approximate nearest neig...
434	허재필	Thanks to compact data representations and fas...	binary code embedding;similarity search;comput...
435	허재필	Many binary code encoding schemes based on has...	binary code encoding;hashing;similarity search...
436	황성재	Java Native Interface (JNI) allows Java applic...	Java Native Interface;native libraries;native ...
437	황성재	Artificial intelligence (AI) is a powerful tec...	Artificial Intelligence;Cybersecurity;Threat D...

438 rows × 3 columns

4. AI / Search Algorithm

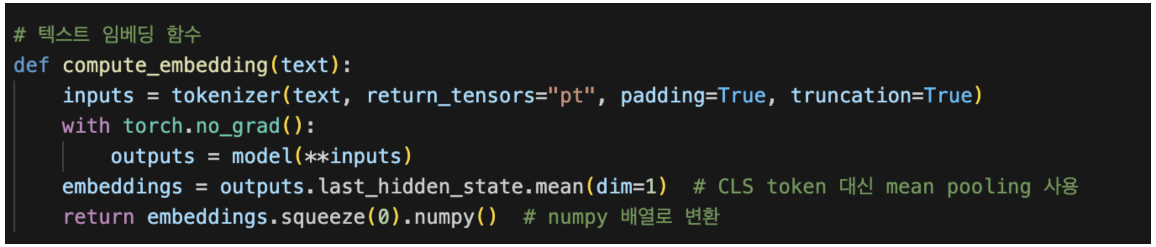
Prof’s feature

1) grouping abstract → Organized by professor



```
# 교수님별 abstract 그룹화
grouped_abstracts = defaultdict(list)
for _, row in data.iterrows():
    grouped_abstracts[row['Name']].append(row['Abstract'])
```

2) mean pooling

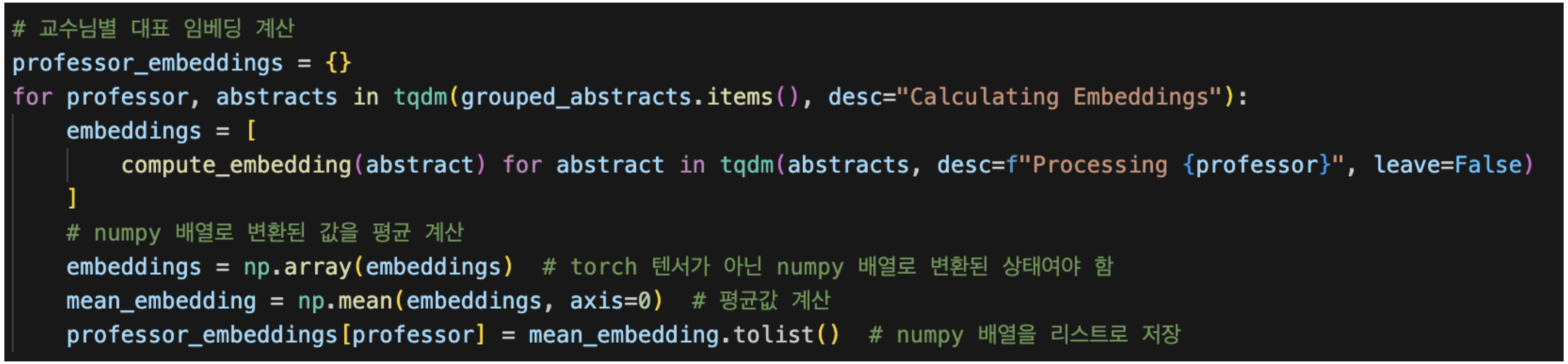


```
# 텍스트 임베딩 함수
def compute_embedding(text):
    inputs = tokenizer(text, return_tensors="pt", padding=True, truncation=True)
    with torch.no_grad():
        outputs = model(**inputs)
    embeddings = outputs.last_hidden_state.mean(dim=1)  # CLS token 대신 mean pooling 사용
    return embeddings.squeeze(0).numpy()  # numpy 배열로 변환
```

4. AI / Search Algorithm

Prof’s feature

3) call the function : Calculating representative embeddings for each professor



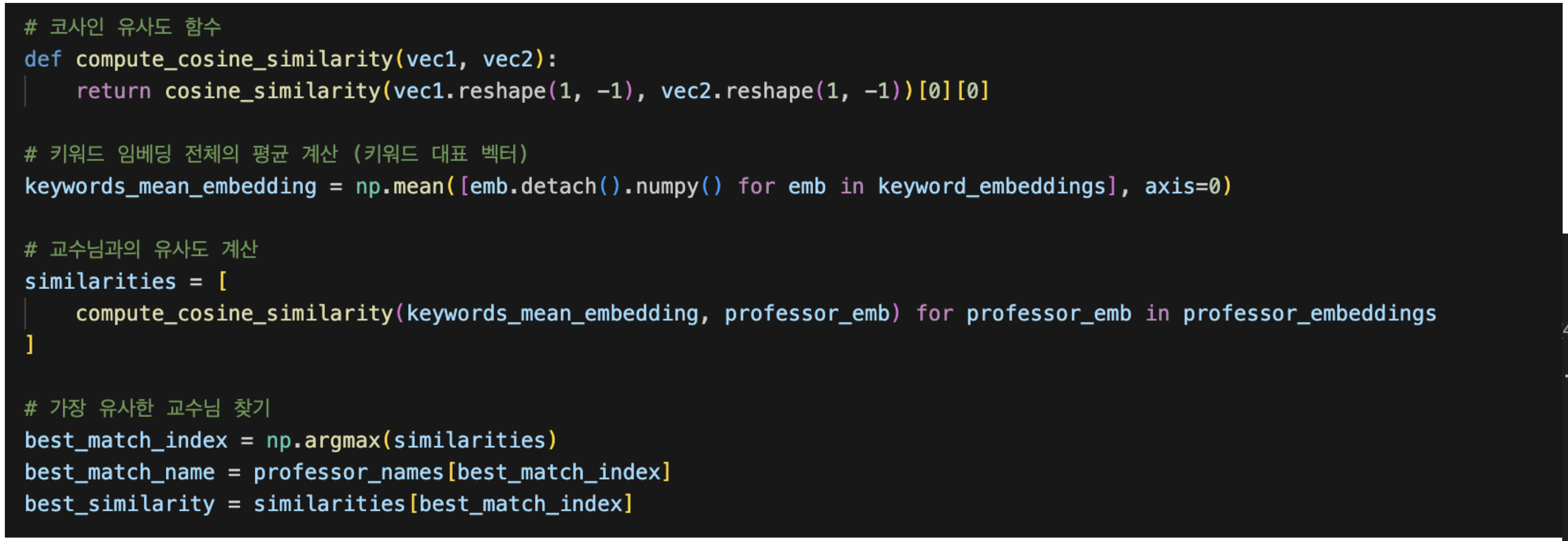
```
# 교수님별 대표 임베딩 계산
professor_embeddings = {}
for professor, abstracts in tqdm(grouped_abstracts.items(), desc="Calculating Embeddings"):
    embeddings = [
        compute_embedding(abstract) for abstract in tqdm(abstracts, desc=f"Processing {professor}", leave=False)
    ]
    # numpy 배열로 변환된 값을 평균 계산
    embeddings = np.array(embeddings)  # torch 텐서가 아닌 numpy 배열로 변환된 상태여야 함
    mean_embedding = np.mean(embeddings, axis=0)  # 평균값 계산
    professor_embeddings[professor] = mean_embedding.tolist()  # numpy 배열을 리스트로 저장
```

time cost: # of 438 abstract, # of 34 prof => about 10minutes

4. AI / Search Algorithm

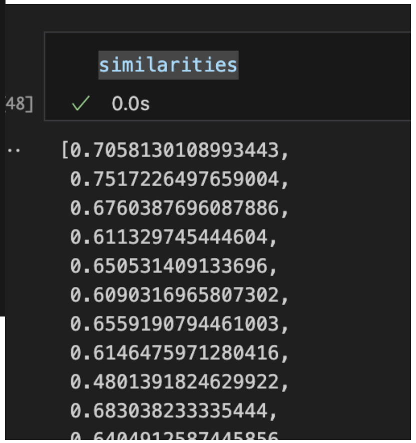
Search

calculate cosine similarity between search keywords and professor embedding



```python
# 코사인 유사도 함수
def compute_cosine_similarity(vec1, vec2):
    return cosine_similarity(vec1.reshape(1, -1), vec2.reshape(1, -1))[0][0]

# 키워드 임베딩 전체의 평균 계산 (키워드 대표 벡터)
keywords_mean_embedding = np.mean([emb.detach().numpy() for emb in keyword_embeddings], axis=0)

# 교수님과의 유사도 계산
similarities = [
    compute_cosine_similarity(keywords_mean_embedding, professor_emb) for professor_emb in professor_embeddings
]

# 가장 유사한 교수님 찾기
best_match_index = np.argmax(similarities)
best_match_name = professor_names[best_match_index]
best_similarity = similarities[best_match_index]
```

<similarity result>

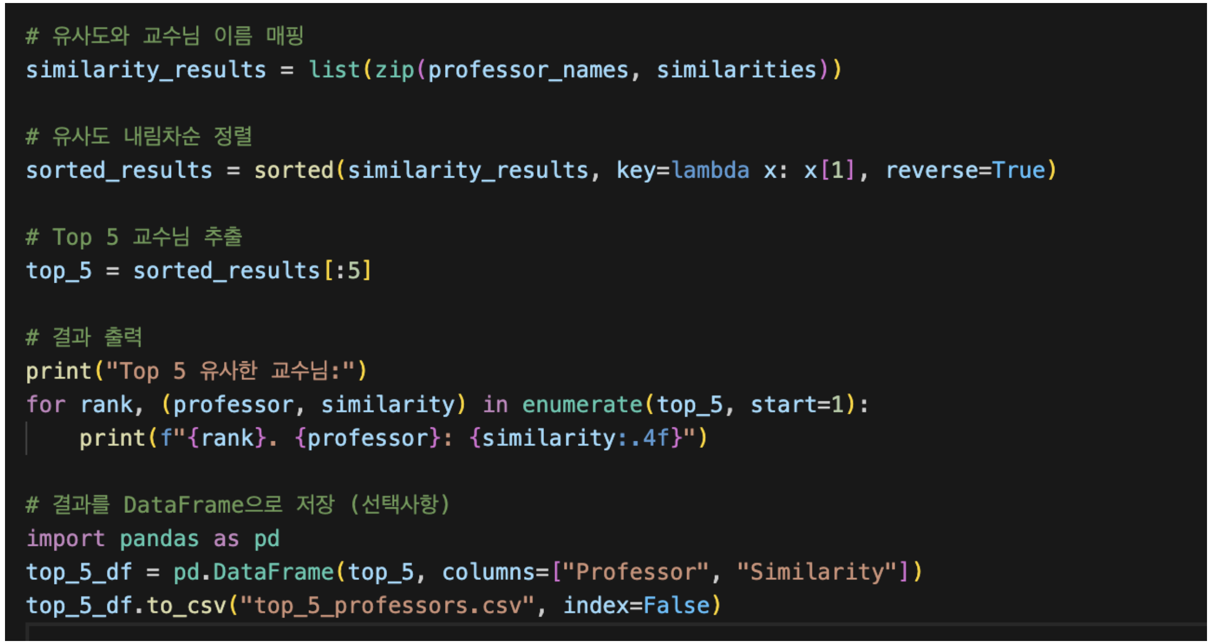


```
    similarities
48]  ✓  0.0s

..  [0.7058130108993443,
     0.7517226497659004,
     0.6760387696087886,
     0.611329745444604,
     0.650531409133696,
     0.6090316965807302,
     0.6559190794461003,
     0.6146475971280416,
     0.4801391824629922,
     0.683038233335444,
     0.6404912587445856
```

4. AI / Search Algorithm

Search top 5

The extracted similarity list is obtained by sorting in descending order.

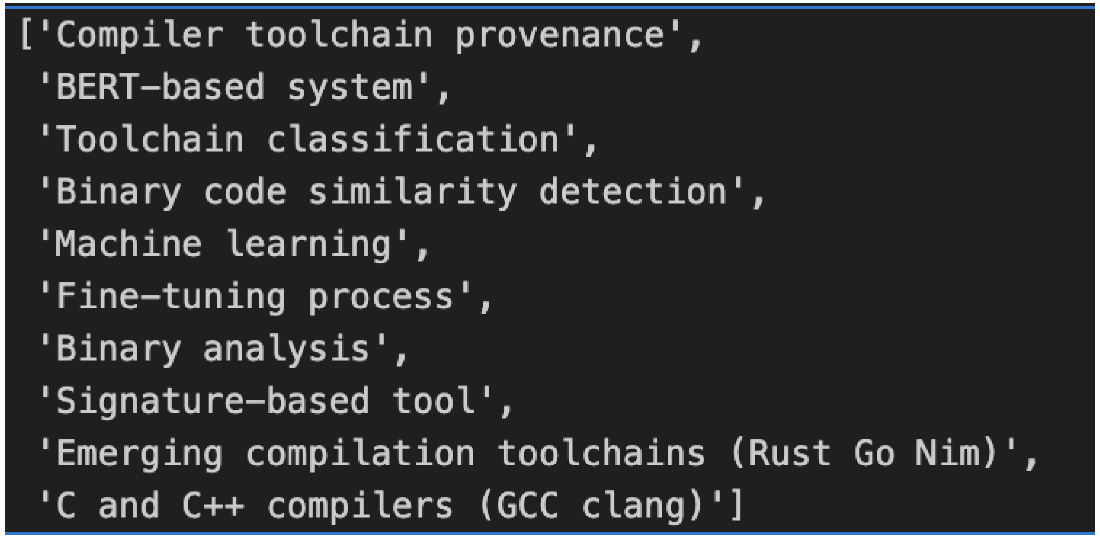


```
# 유사도와 교수님 이름 매핑
similarity_results = list(zip(professor_names, similarities))

# 유사도 내림차순 정렬
sorted_results = sorted(similarity_results, key=lambda x: x[1], reverse=True)

# Top 5 교수님 추출
top_5 = sorted_results[:5]

# 결과 출력
print("Top 5 유사한 교수님:")
for rank, (professor, similarity) in enumerate(top_5, start=1):
    print(f"{rank}. {professor}: {similarity:.4f}")

# 결과를 DataFrame으로 저장 (선택사항)
import pandas as pd
top_5_df = pd.DataFrame(top_5, columns=["Professor", "Similarity"])
top_5_df.to_csv("top_5_professors.csv", index=False)
```

4. AI / Search Algorithm

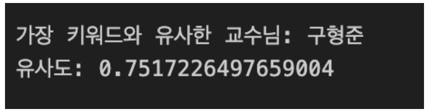
Example

search keywords



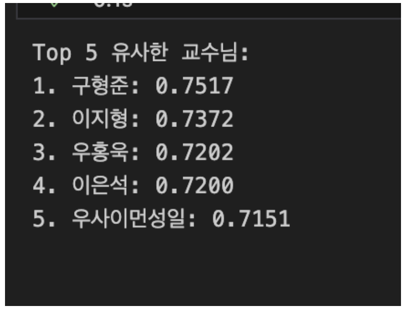
```
['Compiler toolchain provenance',
 'BERT-based system',
 'Toolchain classification',
 'Binary code similarity detection',
 'Machine learning',
 'Fine-tuning process',
 'Binary analysis',
 'Signature-based tool',
 'Emerging compilation toolchains (Rust Go Nim)',
 'C and C++ compilers (GCC clang)']
```

<search result>



```
가장 키워드와 유사한 교수님: 구형준
유사도: 0.7517226497659004
```

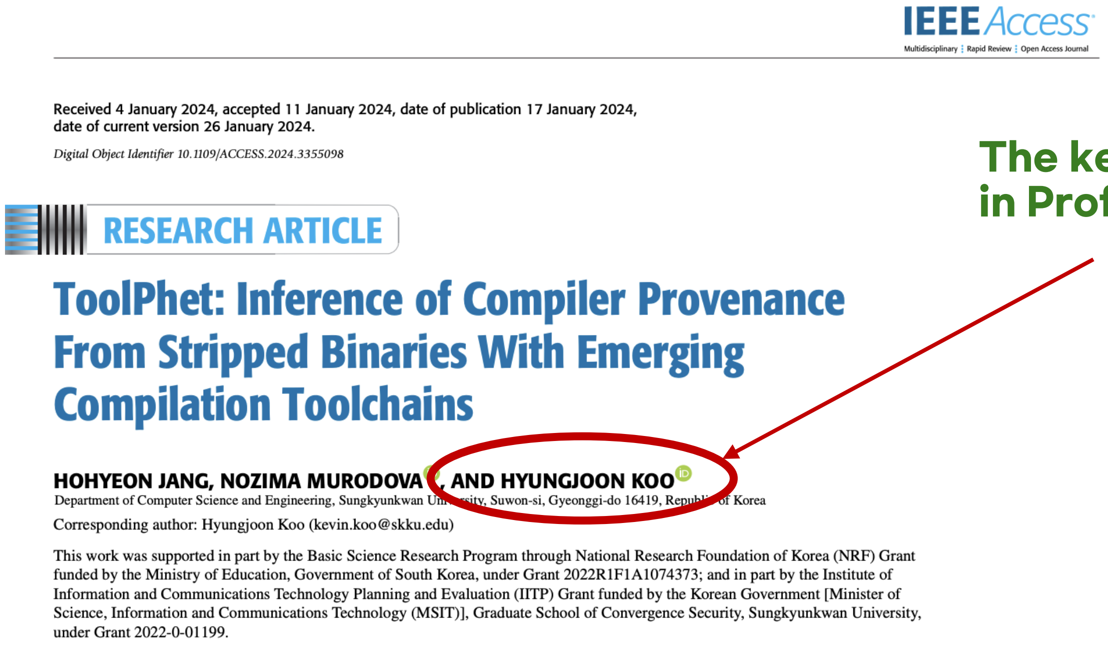
<top 5 search result>



```
Top 5 유사한 교수님:
1. 구형준: 0.7517
2. 이지형: 0.7372
3. 우홍욱: 0.7202
4. 이은석: 0.7200
5. 우사이먼성일: 0.7151
```

4. AI / Search Algorithm

Example



IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 4 January 2024, accepted 11 January 2024, date of publication 17 January 2024, date of current version 26 January 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3355098

RESEARCH ARTICLE

ToolPhet: Inference of Compiler Provenance From Stripped Binaries With Emerging Compilation Toolchains

HOHYEON JANG, NOZIMA MURODOVA, AND HYUNGJOON KOO
Department of Computer Science and Engineering, Sungkyunkwan University, Suwon-si, Gyeonggi-do 16419, Republic of Korea

Corresponding author: Hyungjoon Koo (kevin.koo@skku.edu)

This work was supported in part by the Basic Science Research Program through National Research Foundation of Korea (NRF) Grant funded by the Ministry of Education, Government of South Korea, under Grant 2022R1F1A1074373; and in part by the Institute of Information and Communications Technology Planning and Evaluation (IITP) Grant funded by the Korean Government [Minister of Science, Information and Communications Technology (MSIT)], Graduate School of Convergence Security, Sungkyunkwan University, under Grant 2022-0-01199.

The keywords are highly similar to those used in Professor Koo Hyungjun's actual research papers

Q & A