

기사

# 레시피IS-식재료 인식 기반 레시피 추천 시스템

미구엘 시모에스 로드리게스<sup>1,\*</sup>, 필리페 피달고<sup>1,2</sup> 및 앙헬라 올리베이라<sup>1,2</sup><sup>1</sup> 카스텔로 브랑코 폴리테크닉 연구소, 6000-767 카스텔로 브랑코, 포르투갈<sup>2</sup> 포르투갈 카스텔로 브랑코 폴리테크닉 대학교 디지털 서비스, 애플리케이션 및 콘텐츠 R&D 부서,카스텔로 브랑코, 포르투갈, 6000-767, [ffidalgo@ipcb.pt](mailto:ffidalgo@ipcb.pt) (F.F.); [angelaoliveira@ipcb.pt](mailto:angelaoliveira@ipcb.pt) (O.O.).\* 문의: [msrodrigues105@gmail.com](mailto:msrodrigues105@gmail.com)

**요약:** 현재 음식물 쓰레기는 주로 소비 수준에서 발생하며 환경적, 경제적, 사회적 영향을 미치는 전 세계적인 문제입니다. 음식물 쓰레기 문제를 줄이는 한 가지 방법은 집에 이미 있는 음식을 사용하는 것입니다. 그러나 이것은 특정 음식으로 무엇을 요리해야 하는가라는 또 다른 문제를 야기합니다. 때때로 우리는 어떤 레시피를 만들 수 있는지 모릅니다. 어떤 재료를 섞을 수 있고 어떻게 섞을 수 있는지 아는 것은 초보 요리사에게는 어려운 일이 될 수 있으므로 레시피에 적합한 재료를 선택하는 것이 필수적입니다. 따라서 식재료의 이미지 인식을 통한 레시피 추천 시스템을 개발하고자 합니다. 현재 이 시스템은 사용자가 제공한 이미지를 인식하여 인식된 식재료가 포함된 레시피를 추천하는 웹 애플리케이션입니다. 이를 위해 이미지 인식을 수행할 수 있는 합성곱 신경망 모델인 ResNet-50을 구축하고 약 36종의 채소와 과일이 포함된 데이터셋으로 학습시켰습니다. 이 학습을 통해 모델은 데이터 세트 이미지를 분류하는 데 96%의 정확도를 달성했습니다. 추천 시스템은 인식된 식재료의 라벨을 이용해 레시피를 추출하고, 이 레시피는 에담 API를 통해 검색됩니다.

**키워드:** 식품 성분 인식; 인공 신경망; 데이터 세트; ResNet-50; 추천 시스템

인용: Rodrigues, M.S.; Fidalgo, F.;

Oliveira, Â. 식재료 인식을 기반으로

한 레시피IS-레시피 추천 시스템.

2023, 13, 7880. [https://](https://doi.org/10.3390/app13137880)[doi.org/10.3390/app13137880](https://doi.org/10.3390/app13137880)

아카데미 편집자: 아틸리오 마테라, 프

란체스코 제노베제

수신됨: 2023년 4월 28일

개정일: 2023년 6월 29일

수락되었습니다. 수락: 2023년 7월 3일

발행일: 2023년 7월 5일

**저작권:** © 2023 작성자. 라이선스 소유자 MDPI, 스위스 바젤. 이 문서는 크리에이티브 커먼즈 저작자표시(CC BY) 라이선스 조건에 따라 배포되는 오픈 액세스 문서입니다(<https://creativecommons.org/licenses/by/4.0/>).



## 1. 소개

현재 사용자 프로파일 이 있는 여러 웹사이트 또는 모바일 애플리케이션을 찾을 수 있습니다. 이러한 웹사이트나 애플리케이션은 사용자가 최초 등록 시 제공한 선호도나 취향 등의 정보를 기반으로 하는 추천 시스템을 사용합니다. 요리 업계에서는 재료 이름, 요리 유형 또는 문화와 같은 키워드

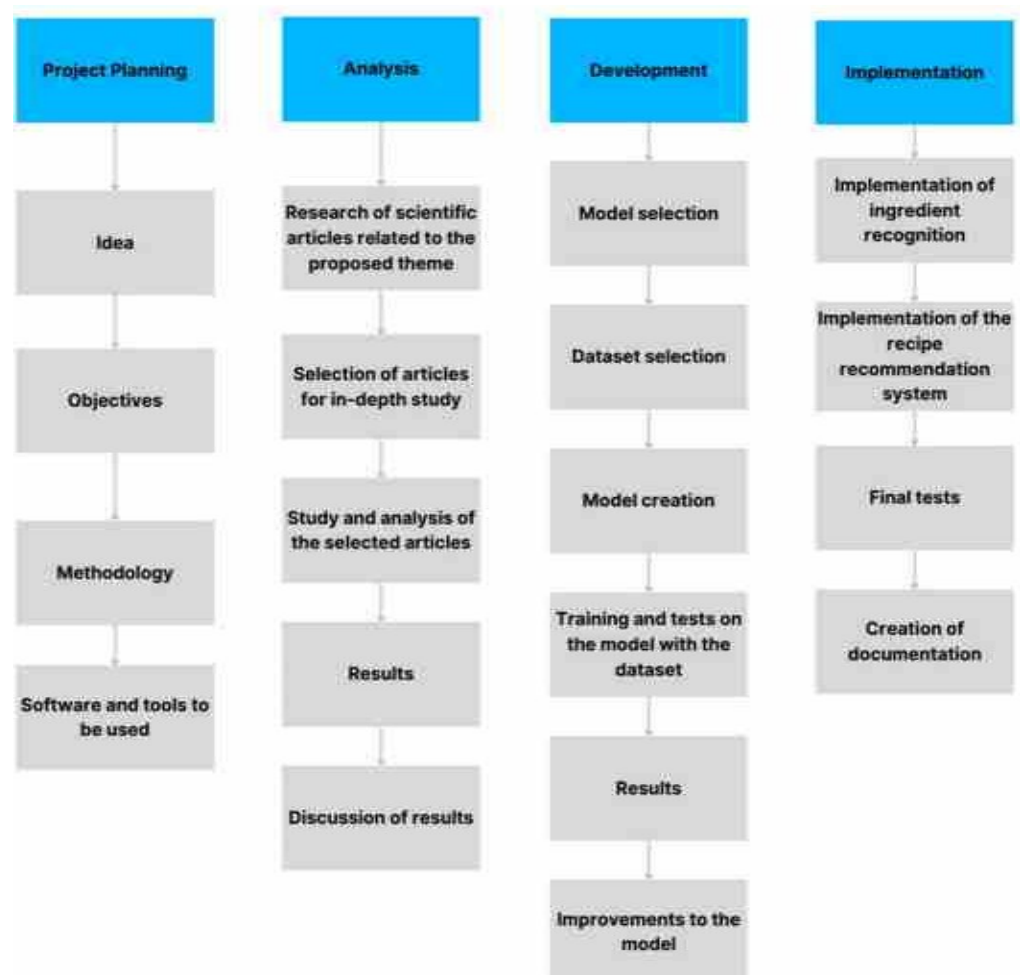
를 기반으로 레시피를 찾는 데 사용되는 여러 앱과 웹사이트가 있습니다. 이러한 애플리케이션은 사용자의 요구와 관심사를 고려하지만 재료의 최대 개수 제한과 같은 제한 사항을 파악하지 못합니다. 이러한 경우 사용자는 결국 재료를 누락하거나 비슷한 재료로 대체하게 됩니다. 사용자가 이러한 조정을 피할 수 있도록 재료 인식을 사용하여 사용자가 그 순간에 사용할 수 있는 재료를 식별하고 해당 재료 또는 해당 재료를 통해 다양한 레시피를 추천할 수 있습니다. 식재료 인식을 통한 레시피 추천 시스템의 가장 큰 목적은 집에 있는 식재료를 요리할 하고자 하는 사용자가 음식물 쓰레기를 줄일 수 있도록 돕는 것입니다. 이 시스템은 가정뿐만 아니라 마트에서도 스마트폰으로 식재료를 식별하여 사용할 수 있습니다. 이렇게 하면 사용자는 시스템의 추천을 바탕으로 일주일 동안 무엇을 요리할지 계획하거나 아이디어를 만들 수 있습니다. 음식 이미지 인식이나 물체 인식에는 텐서플로우[1], 케라스[2], 넘피[3]와 같은 딥러닝 모델과 라이브러리가 사용됩니다. 이러한 라이브러리를 사용하면 ResNet, MobileNet, VGG16 등과 같이 사전 학습된 딥러닝 모델을 구현할 수 있습니다. 다음으로, 딥러닝 모델이 데이터 세트가 제공하는 입력 이미지로부터 학습할 수 있도록 데이터 세트를 사용하여 딥러닝 모델을 훈련시켜야 합니다. 이

방식으로, 모델은 이미지의 패턴을 인식하여 최종 단계에서 이미지를 분류할 수 있습니다.

따라서 식재료 인식 문제를 해결하기 위해 ResNet-50 모델을 사용할 것을 제안합니다. 또한 재료 이미지가 포함된 데이터 세트가 적용되어 모델에 입력 이미지가 제공됩니다. 그런 다음 ResNet-50 모델 아키텍처의 마지막 부분에 일부 컨볼루션 레이어를 추가하여 정확도를 높입니다. 이러한 변경을 통해 식재료 분류에서 96%의 정확도를 얻을 수 있었습니다.

ResNet-50을 통합하여 시각적 특성을 기반으로 식재료를 효과적으로 분석하고 분류할 수 있었습니다. 이 모델은 성분을 정밀하게 식별하는 데 탁월했기 때문에 정확하고 신뢰할 수 있는 추천을 제공할 수 있었습니다. 또한, 이 프로젝트는 포괄적이고 광범위한 요리 정보 저장소인 Edamam API 데이터베이스와 원활하게 통합되었습니다. 이 API의 기능을 활용하여 방대한 레시피, 영양 데이터, 요리 기법 모음에 액세스할 수 있었기 때문에 새로운 데이터베이스를 개발할 필요가 없었습니다.

프로젝트의 구조와 진행 상황을 포괄적으로 이해하기 위해 그림 1에 표시된 순서도를 작성하여 이 작업의 작업을 설명했습니다. 이 시각적 표현은 프로젝트 계획, 분석, 개발 및 구현의 단계를 식별합니다.



---

**그림 1.** 프로젝트 단계별 순서도.

본 논문은 다음과 같이 구성되어 있습니다: 섹션 2에서는 체계적인 개정 방법론에 의해 뒷받침되는 연구 방법을 제시합니다. 섹션 3에서는 프로젝트 개발에 사용된 자료와 방법을 소개하고, 제안된 머신러닝 모델과 그 아키텍처, 사용된 데이터 세트, 그리고 데이터 세트와 함께 모델을 학습하는 과정을 소개합니다.

섹션 4에서는 프로젝트의 전체 구현을 두 부분으로 나누어 첫 번째는 식재료 인식 설명 구현, 두 번째는 인식된 식재료 구현을 기반으로 한 레시피 추천으로 나누어 발표합니다. 마지막으로 섹션 5에서는 향후 작업 구현에 대한 결론과 제안을 공유합니다.

## 2. 관련 작품

음식, 재료, 레시피 추천에 대한 인식은 시간이 지남에 따라 인기를 얻고 있습니다. 머신러닝 방법은 인공 합성곱 신경망을 활용하여 이미지의 패턴을 인식하고 분류합니다. 연구를 통해 식재료 인식과 레시피 추천에 대해 논의하는 여러 논문이 발견되었습니다. 이 체계적 문헌고찰은 PRISMA 방법론(체계적 문헌고찰 및 메타분석을 위한 선호 보고 항목)을 따랐습니다[4]. 다음 주제가 포함되었습니다:

- A. 연구 질문
- B. 포함 기준
- C. 연구 전략
- D. 결과
- E. 데이터 추출 및 분석
- F. 토론

연구를 수행하기 위해 음식 이미지 인식과 레시피 추천을 주제로 연구 질문을 공식화했습니다. 이러한 질문은 식별된 문제에 대한 해결책을 제안하는 것을 목표로 합니다. 우리가 사용한 연구 질문은 다음과 같습니다:

- 질문 1: 식재료 인식 또는 식재료 이미지를 기반으로 레시피 제안을 제공하는 디지털 솔루션이 있나요?
- 질문 2: 성분 인식 솔루션의 프레임워크는 무엇인가요?
- 질문 3: 음식물 쓰레기를 줄이기 위해 남은 음식을 기반으로 레시피를 제안할 수 있는 디지털 솔루션이 있나요?

포함 기준은 재검색자가 연구 질문에 답하는 데 사용할 대상 집단의 주요 특성을 나타냅니다. 본 연구에 정의된 포함 기준은 다음과 같습니다:

- 기준 1: 2014년부터 2021년까지 연구;
- 기준 2: 영어로 작성된 연구;
- 기준 3: 식재료 또는 식재료 이미지 인식을 기반으로 레시피를 제안하는 연구;
- 기준 4: 식재료의 이미지 인식에 관한 연구;
- 기준 5: 전체 텍스트가 제공되는 연구.

IEEE Xplore [5], ACM Digital [6], ScienceDirect [7] 데이터베이스를 사용하여 관련 논문을 검색했습니다. 검색어에는 "성분 인식" 및 "애플리케이션", "패키지" 및 "포장 식품" 및 "음식 인식", "프레임 작품" 및 "레시피 추천", "음식물 쓰레기 레시피" 및 "애플리케이션" 등이 포함되었습니다. 2021년 11월에서 12월 사이에 검색했습니다.

기준 1을 적용한 결과, 그림 2에 제시된 바와 같이 총 146개의 과학 연구가 발견되었으며, 이 중 85개는 IEEE Xplore에서, 46개는 ACM Digital에서, 15개는 ScienceDirect에서 검색되었습니다. 그런 다음 기준 2를 적용하여 중복을 제거한 결과 130개의 연구가 남았습니다. 이 연구들을 기준 3과 4에 따라 분석했습니다. 그런 다음 기준 3, 4, 5를 적용하여 나머지 79개 연구에 대한 전체 텍스트 분석을 수행했습니다. 이러한 기준에 따라 16개의 연구가 검토 대상에 포함되었습니다.

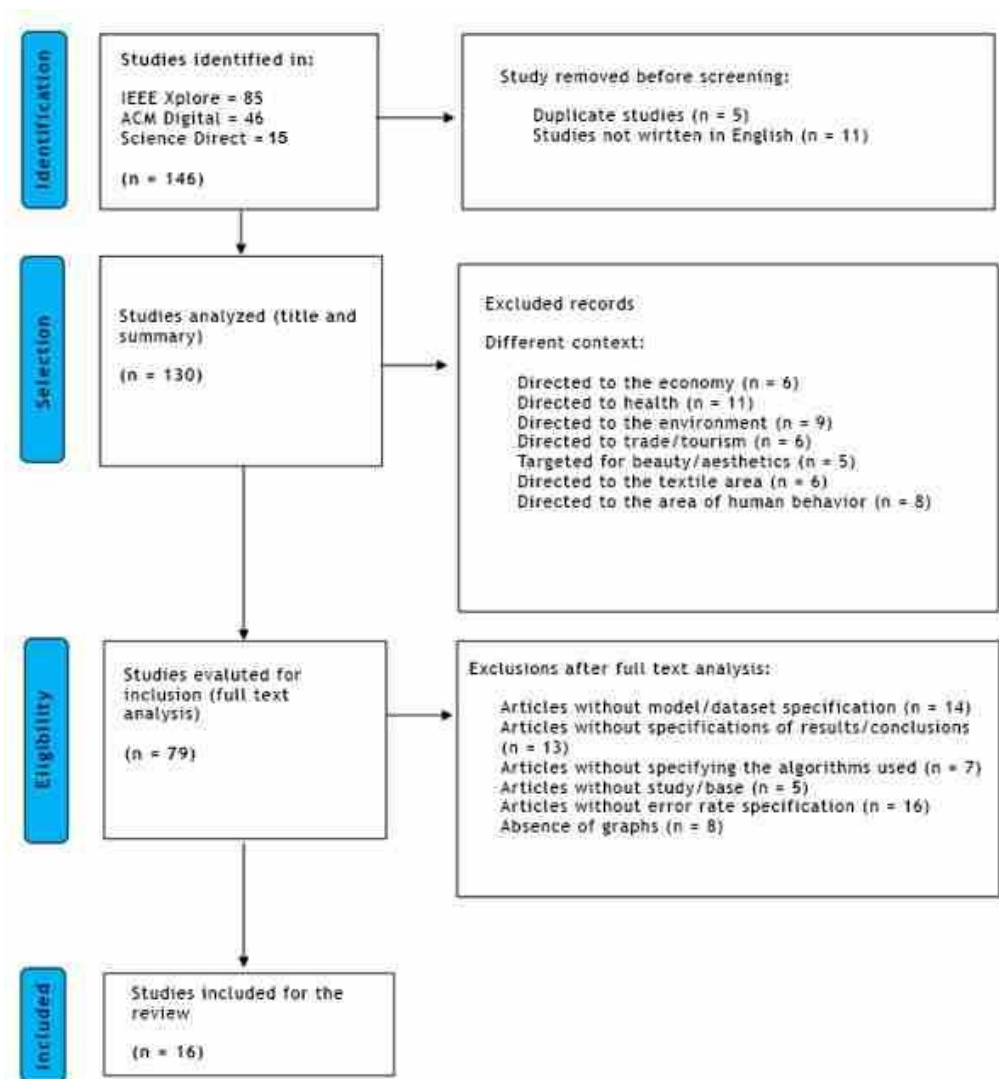


그림 2. 연구 단계의 순서도.

데이터는 사전 정의된 형식을 사용하여 식별된 모든 연구에서 추출되었습니다. 추출된 데이터에는 연구, 모델, 설명이 포함되었습니다. 표 1은 포함된 연구에서 추출된 데이터를 식별합니다.

표 1. 분석된 과학 논문.

연구	모델	설명
[8]	인셉션-v3 인 셉션-v4	저자들은 모바일 기기의 광범위한 사용을 활용하여 섭취한 음식에 대한 건강 정보를 제공하는 시스템을 개발했습니다. 이 애플리케이션은 식사 인식을 기반으로 하며, 식재료와 영양가를 표시합니다.
[9]	ResNet-50 Inception V3 SENet-154	저자는 레시피의 특정 재료에 대한 이미지 수준에서의 인식, 즉 인식된 레시피에 의해 구성된 재료의 검출, 단일 또는 다중 이미지 스케일링에서의 그룹화와 같은 몇 가지 문제 분석에 중점을 두고 재료 및 레시피를 인식하는 시스템을 개발했습니다.
[10]	AlexNet 카페넷 레 스넷	저자들은 여러 고급 머신 러닝 기법을 적용하여 다중 클래스 분류의 평균 정확도를 향상시키는 머신 러닝을 사용하여 성분 이미지 데이터 세트에서 효과적인 특징을 추출하는 프레





표 1. 계속

연구	모델	설명
[11]	MatConvNet	저자는 음식 인식을 공동으로 고려하기 위해 신경망을 사용할 것을 제안합니다, 성분 인식, 제조 방법 인식 등 여러 요소를 고려하여 인식 성능을 개선할 수 있는지 확인합니다.
[12]	ResNet 101 NeuMF	저자는 공간 정규화 네트워크(SRN) 모델을 사용하여 슈퍼마켓에서 여러 식재료를 인식하는 솔루션을 제안하고 이를 해결합니다. 또한 여러 식재료 인식과 결합하여 개인화된 레시피를 추천하는 방법도 제시합니다.
[13]	ResNet-50	저자는 레시피를 인식하는 시스템을 제안하고 해당 이미지에서 전체 과정과 준비 및 재료에 대한 지침을 포함하여 레시피를 요리하는 과정을 제시합니다.
[14]	ResNet-50	저자는 요리 레시피와 음식 이미지 간의 교차 모달 인식 작업을 분석하고 음식 영역에서 인식 작업을 해결하기 위해 적대적 교차 모달 임베딩(ACME) 프레임워크를 제안했습니다. 주요 목표는 요리 레시피와 음식 이미지 사이의 엔드투엔드 ACME 프레임워크를 만드는 것이었습니다.
[15]	ResNet DenseNet	저자는 각 레시피의 재료를 감지하고 캡처하는 것을 목표로 하는 음식 인식을 위한 다중 스케일 다중 뷰 특징 집계(MSMVFA) 체계를 제안합니다. MSMVFA 방법은 지도 합성곱 신경망(CNN)을 사용하여 각 유형의 재료를 식별한 다음 이를 융합하여 인식된 레시피를 보다 강력하고 차별적으로 표현합니다. 제안된 방법의 효과와 견고성을 검증하기 위해 저자들은 세 가지 모델을 구현했습니다: VGG, ResNet, DenseNet.
[16]	RESNet-50 SE-ResNet-50 CBAM-ResNet-50	아홉 번째 기사의 저자는 애플리케이션을 통해 조리법이나 음식을 인식하는 데 어려움이 있다고 언급합니다. 따라서 이 글에서 저자는 재료 인식과 레시피 생성 모두에서 음식 인식 작업에 여러 의미적 특징을 통합하는 멀티태스크 프레임워크 내에서 멀티뷰 주의 네트워크(MVANet)를 제안합니다. 또한 저자는 여러 작업에서 여러 의미적 특징을 추출하고 융합하는 역할을 하는 새로운 멀티뷰 주의집중 융합(MVAF)을 제안합니다. 구현 측면에서 저자는 MXNet 소프트웨어에서 심층 신경망을 사용하여 ResNet-50, E-ResNet-50을 포함한 모든 테스트 모델을 구현했습니다, CBAM-ResNet-50 및 저자의 모델 MVANET50.
[17]	VGG Arch-D AlexNet	저자는 성분을 인식하고 성분 인식과 식품 분류를 동시에 수행하기 위한 아키텍처를 제안하고, 이들 간의 상호 관계를 탐구합니다. 아키텍처 측면에서 저자는 Arch-A, Arch-B, Arch-C, Arch-D라는 네 가지 딥러닝 아키텍처를 사용하여 DCNN 모델을 수정했습니다. 이러한 아키텍처는 16 계층에서 수정되었습니다. VGG 모델.
[18]	VGG	저자들은 Pic2Dish라는 모바일 애플리케이션을 개발했습니다.
[19]	RESTful	[20] RESNet-50
		[21] 클라우드 브로커

2023. 13. 7880	는 요리 이름을 인식하고 요리 방법에 대한 교육용 동영상과 함께 맞춤형 레시피를 추천합니다. 인식 모듈에는 VGG 모델을 통한 컨볼루션 네트워크를 사용했습니다.	는 사용자 인터페이스, 이미지 인식 모델, 검색 및 추천 엔진, 데이터 세트의 네 가지 모듈 <sup>936</sup> 에 의존합니다. 이 아키텍처를 구현하기 위해 RESTful 서비스가 사용됩니다.
저자는 요리 작업을 단순화하기 위해 딥러닝을 사용하는 시스템 설계 및 구현을 제안합니다. 시스템 아키텍처	저자는 주어진 레시피에서 식재료의 검출을 통해 영양 정보를 추정하는 방법을 제안합니다. 저자는 이미지 결과를 기반으로 식재료 텍스트 태그의 발생 빈도를 표현하는 라벨 히스토그램을 추출한 후, 이전 정보를 기반으로 회귀 분석을 통해 영양가를 추정하는 의미적 분할을 통해 영양가를 추정하는 모델을 제시합니다.	저자는 음식 이미지를 분류하고 각 음식의 칼로리를 높은 정확도로 계산하는 클라우드 기반 모바일 eHealth 칼로리 시스템을 제안합니다. 이 논문에서는 클라우드에서 관리되는 인스턴스를 효율적으로 사용하기 위해 지능형 클라우드 브로커 메커니즘을 적용하여 더 나은 성능과 더 빠르고 효율적인 응답 시간을 제공한다는 점이 특이합니다.

표 1. 계속

연구	모델	설명
[22]	AlexNet	이 글의 저자는 음식 이미지를 감지하고 인식하기 위해 컨볼루션 신경망(CNN)을 사용했습니다. 그들은 CNN의 하이퍼파라미터를 최적화하고 은 서포트 벡터 머신(SVM)을 사용하는 기존 방식에 비해 음식 인식의 정확도가 크게 향상되었음을 입증했습니다. 저자들은 생성된 CNN을 관찰한 결과, 색상 특징이 음식 인식 프로세스에서 지배적인 역할을 한다는 사실을 발견했습니다.
[23]	Inception-v3	저자들은 개인 맞춤형 건강 식품 추천 체계인 'Market2Dish'라는 건강 식품 추천 체계를 제시합니다. 음식 이미지 캡처에는 Inception-v3 Net이 사용되었으며, 95.5%의 정확도를 달성했습니다. 건강 프로파일링 측면에서 저자는 소셜 네트워크에서 동일한 사용자가 생성한 콘텐츠를 통해 각 사용자의 건강 프로필을 추적하는 지도 뉴런 모델을 제안합니다.

선정된 논문을 분석한 결과, 특정 음식이나 레시피를 인식하는 데 사용할 수 있는 다양한 도구와 솔루션이 있다는 것이 분명해졌습니다. 검토한 연구에서 가장 중요한 측면 중 하나는 건강한 식습관이 사람들의 웰빙에 점점 더 중요한 역할을 하고 있기 때문에 더 건강한 음식과 레시피를 식별하거나 추천하기 위해 고안된 모델을 개발하거나 검토하는 것입니다.

분석 대상 연구의 기간은 그림 3에서 볼 수 있듯이 2014년부터 2021년 사이에 수행되었으며, 대부분 2017년과 2019년에 발표되었습니다.

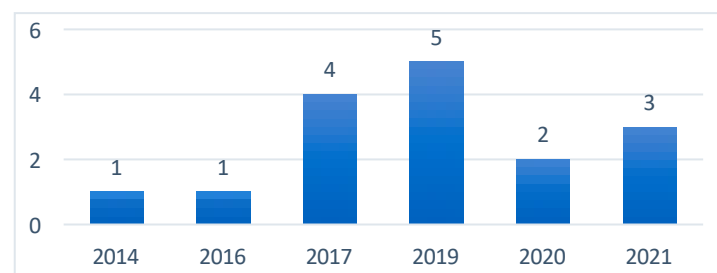


그림 3. 기사가 게시된 날짜.

그림 4에서 볼 수 있듯이 이러한 연구의 저자들은 유사한 아키텍처를 구현하는 것을 선호했으며, 94%의 논문이 컨볼루션 신경망(CNN)을 활용하고 6%만이 다른 딥 러닝 방법을 사용했습니다. CNN은 자동화된 엔드투엔드 예측을 위해 설계된 딥 러닝 모델입니다. 따라서 분석 대상 연구에 사용된 모든 아키텍처는 CNN 모델을 사용했으며, 각 기사마다 특정 목적에 따라 다른 네트워크가 사용되었다는 결론을 내릴 수 있습니다.

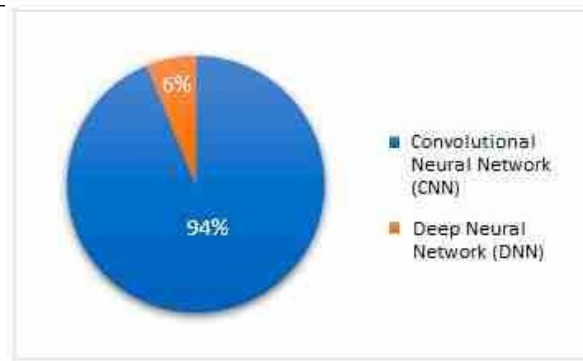


그림 4. 기사에서 사용된 아키텍처의 비율.

그림 5에서 볼 수 있듯이 각 연구를 구현하는 데 사용된 모델은 매우 다양했는데, 어떤 저자는 한 가지 모델만 사용하기로 선택한 반면 다른 저자는 어떤 모델이 가장 좋은지 보여주기 위해 여러 모델을 구현하는 것을 선호했기 때문입니다.

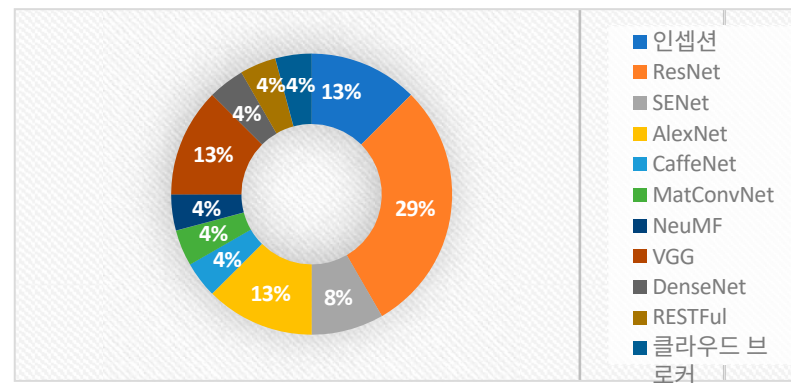


그림 5. 기사에서 사용된 아키텍처의 비율.

모델 분석 및 기능 테스트를 위해 분석된 연구에 사용된 데이터 세트의 경우, 그림 6에서 볼 수 있듯이 저자들은 서로 다른 대안을 사용했습니다. 일부 저자는 구현된 모델을 테스트하고 얻은 결과를 검증 및 비교하기 위해 두 개 이상의 데이터 세트를 사용했습니다. 일부 작성자는 구현된 모델을 테스트하고 결과를 비교하기 위해 여러 데이터 세트를 활용했습니다. 가장 일반적으로 사용된 데이터 세트는 많은 저자가 식품 카테고리 측면에서 포괄적이고 완전한 것으로 간주하는 VIREOFood였습니다.

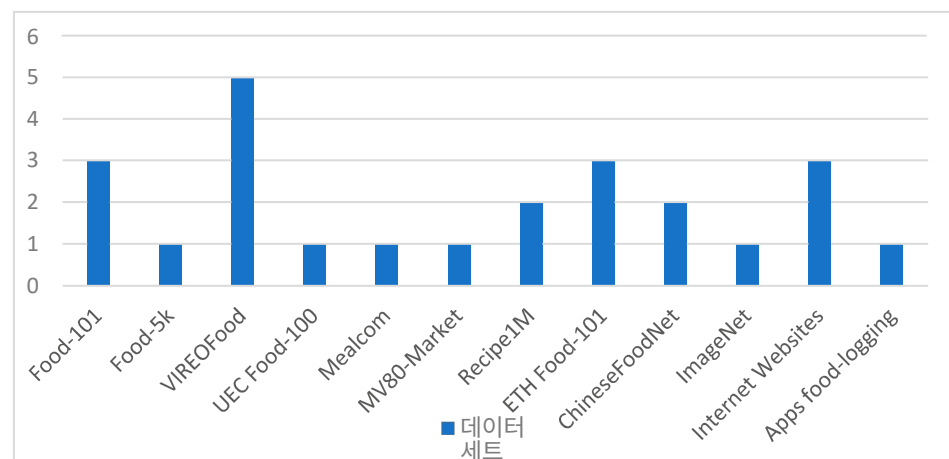


그림 6. 기사에 사용된 데이터 세트의 수량.

논문 중 3편은 웹사이트를 통해 사용자로부터 음식이나 레시피에 대한 정보와 이미지를 수집했고, 1편은 푸드 로깅 앱을 사용해 스마트폰의 고화질 이미지를 데이터셋으로 만들었다는 점에 주목할 필요가 있습니다. 결론적으로, 본 연구를 통해 음식 인식 및 추천에 다양한 모델이 활용될 수 있으며, 컨볼루션 신경망이나 딥러닝을 사용하여 구현할 수 있음을 확인할 수 있었습니다. 그러나 구현에 사용된 소프트웨어에 대한 정보를 제공한 논문은 케라스(Keras), 파이토치(PyTorch), MXNet 등 세 편에 불과했습니다.

이 주제에 대한 많은 논문이 있지만, 대부분의 논문은 연구와 개발 방식이 매우 유사합니다. 연구와 개발의 유사성이 많지만 결과가 거의 없거나 사용 된 개발 방법에 대해 언급하지 않은 논문이 많이 있습니다. 이는 논문의 내용을 분석해 보면 알 수 있습니다. 146개의 논문 중 17개(16개)만이 결과와 중요한 내용을 담고 있으며, 의도한 주제 범위 내에 포함된 것으로 나타났습니다.

과학 논문을 분석하고 그 결과를 검증 한 결과, 구현 방법이 가장 중요 할뿐만 아니라 사용할 소프트웨어, 아키텍처 및 주어진 음식을 인식하고 레시피를 추천하는 데 가장 정확한 모델을 잘 선택하는 것이 기본이라는 결론을 내 렸습니다. 또한 레시피 인식 및 추천을 수행하기 위해 다양한 유형의 모델과 아키텍처를 사용할 수 있으며, 이러한 구현은 아키텍처나 모델을 변경해야 하는 경우가 있기 때문에 기존 프로그래밍으로 해결하기에는 너무 복잡한 프로그래밍이 필요하다는 것을 알 수 있습니다.

위의 정보를 바탕으로 연구 질문에 대한 답변이 제시됩니다:

- (a) 질문 1: 식재료 인식 또는 식재료 이미지를 기반으로 레시피 제안을 제공하는 디지털 솔루션이 있나요?

제안된 주제가 포괄적이고 식품 산업에서 이미지 인식이 점점 더 보편화됨에 따라 많은 사람들이 작업을 단순화하기 위해 모바일 애플리케이션을 사용합니다. 연구된 과학 논문 중 [8-23] 논문에서 참조한 바와 같이 다양한 구현 방법을 사용하여 성분 인식을 위한 다양한 솔루션이 발견되었습니다. 식품 패키지 인식에 대해 구체적으로 언급한 논문은 없지만, 일부 논문에서는 VireoFood 데이터 세트와 같이 포장된 식품의 이미지가 포함된 이미지 데이터 세트를 테스트했습니다. 이 데이터 세트에는 레시피, 음식, 재료 및 일부 포장 식품의 사진이 포함되어 있으며 [9,15-18] 논문에서 언급되었습니다.

- (b) 질문 2: 성분 인식 솔루션의 프레임워크는 무엇인가요? 프레임워크는 일련의 일반 코드가 포함된 인터페이스로, 개발자가 이를 사용할 수 있도록 지원합니다. 모델을 빠르고 효과적으로 구현하여 특정 기능을 재현하는 데 필요한 시간을 단축할 수 있습니다. 식품 또는 성분 인식이라는 의도된 주제와 관련하여, 문서 [13]에서 언급된 PyTorch, 문서 [16]의 MXNet, 문서 [19]의 Keras와 같이 사용 가능한 여러 프레임워크가 이 기능을 수행할 수 있습니다.

- (c) 질문 3: 음식물 쓰레기를 줄이기 위해 남은 음식을 기반으로 레시피를 제안할 수 있는 디지털 솔루션이 있나요?

음식물 쓰레기는 높은 소비량과 다양성으로 인해 심각한 사회적, 경제적, 환경적 결과를 초래하는 오늘날의 세계에서 점점 더 큰 문제로 대두되고 있습니다. 여러 기사를 분석했지만 음식물 쓰레기 문제를 직접적으로 다룬 기사는 없었습니다. 그러나 음식물 쓰레기를 줄이기 위해서는 유통기한 내에 소비할 수 있는 식품을 구매하는 것과 같은 행동이 포함될 수 있으며, 구매한 식품의 인식을 기반으로 레시피를 추천하는 것도 한 가지 방법입니다. 식품 또는 재료 인식에 기반한 레시피 추천에 초점을 맞춘 논문은 [12,18,19,23]입니다.

종합적인 연구 결과, 분석된 과학 논문에서 가장 자주 사용되고 언급된 것은 ResNet-50 모델이라는 결론을 내렸습니다. 또한, Food-101 데이터 세트는 광범위하고 주로 여러 성분이 포함된 이미지로 구성되어 있고 단순 성분의 이미지가 적음에도 불구하고 테스트 및 결과에 가장 많이 인용되고 활용되었습니다.

### 3. 자료 및 방법

작업은 두 부분으로 나뉩니다. 첫 번째 파트는 주어진 데이터 세트를 통해 식재료를 인식하는 것을 목표로 합니다. 두 번째 파트는 첫 번째 파트에서 인식한 식재료를 기반으로 레시피를 추천하는 것입니다.

#### 3.1. 성분 식별을 위한 제안 모델

식품 성분 식별의 첫 번째 부분에서는 컨볼루션 신경망(CNN) 모델을 만들었습니다. 이 경우 과학 논문에서 자주 사용되고 주어진 이미지의 분류 또는 인식이 목적인 때 가장 일반적으로 사용되는 모델 중 하나이기 때문에 사전 학습된 모델인 ResNet-50이 사용되었습니다. ResNet-50 모델은 원래 ResNet 모델의 변형 모델입니다. 두 모델 모두 동일한 처리 개념을 사용하지만 레이어 수에 차이가 있습니다. ResNet-50은 네트워크에 50개의 레이어를 사용하는 반면, ResNet 모델은 34개의 레이어를 사용합니다. ResNet의 아키텍처



50 모델은 5개의 블록인 컨볼루션 블록 시퀀스로 구성됩니다[24]. ResNet-50 모델은 48개의 컨볼루션 레이어로 구성되며, MaxPool 레이어 1개와 AveragePool 레이어 1개도 있습니다.

처음에 프로젝트는  $224 \times 224$  또는  $128 \times 128$  크기의 입력을 받습니다. 이 구현에서는  $128 \times 128$  입력이  $224 \times 224$  입력보다 메모리와 처리 시간이 덜 필요하기 때문에  $128 \times 128$  입력이 사용됩니다. 전체 ResNet 아키텍처에는 컨볼루션 및 최대 풀링 레이어와 같은 초기 레이어가 포함됩니다. 다음으로 신경망은 세 개의 레이어 블록을 포함하는 첫 번째 상태인 상태 1로 시작됩니다. 신원 블록[25]은 ResNet 신경망에서 사용되는 표준 블록입니다. 입력값이 출력값의 크기와 동일한 경우에 해당합니다. 크기가 다르면 컨볼루션 블록을 사용하여 입력값을 출력값과 동일하게 만듭니다. ResNet 모델은 원래 입력값에 컨볼루션 블록의 출력값을 더하는 스킵 연결[26]이라는 개념을 사용합니다. 이 건너뛰기 연결은 ResNet 모델에서 매우 중요한 개념입니다[25].

각 상태 전환이 발생할 때마다 창 크기는 두 배가 되고 입력 크기는 절반으로 줄어듭니다. 마지막으로 네트워크에는 평균 풀링 레이어와 소프트맥스 활성화 기능이 있는 완전 연결 레이어, 분류를 위한 1000개의 뉴런이 포함됩니다. ResNet-50 모델에는 약 2,300만 개의 훈련 가능한 파라미터가 있습니다. 또한 모델의 정확도 수준을 높이기 위해 두 개의 컨볼루션 신경망 레이어가 추가되었으며, 이는 선택한 데이터 세트를 사용하여 테스트 단계에서 더욱 정교해질 것입니다.

사용된 배치 크기는 32개로, 이는 각 훈련 반복 중에 모델이 한 번에 32개의 훈련 예제를 사용하여 매개변수를 처리하고 업데이트한다는 의미입니다. 이 배치 크기를 선택한 이유는 일반적으로 배치 크기 32가 계산 사용 측면에서 가장 효율적이기 때문에 많은 메모리와 처리를 필요로 하지 않기 때문입니다. 최적화 기법으로는 가장 널리 사용되는 기법 중 하나이며, 모델의 각 파라미터에 대해 개별적으로 학습률을 조정하여 기울기의 평균에 따라 조정하기 때문에 "adam" 최적화 기법을 선택했습니다. 학습률과 관련하여 "adam" 옵티마이저가 사용되었으므로 이 옵티마이저의 학습률은 0.001 값으로 설정되어 있습니다. 이 학습 속도는 빠른 수렴과 안정성 사이의 적절한 균형을 제공합니다.

### 3.2. 데이터 세트

데이터 세트는 인공지능 알고리즘이나 다른 유형의 데이터 과학 프로젝트를 훈련하기 위한 샘플 역할을 하는 특정 데이터베이스를 말합니다. 올바른 데이터셋을 선택하는 것은 정확한 결과를 생성하는 모델을 훈련하는 데 매우 중요합니다. 유사한 주제에 대한 과학적 연구를 분석한 결과, Food-101 데이터 세트가 인기 있는 선택이라는 점이 강조되었습니다. 그럼에도 불구하고 이 데이터 세트는 약 101,000개의 이미지로 구성되어 있으며 대부분 조리된 레시피 이미지가 포함되어 있어 간단한 재료를 사용하려는 프로젝트의 주요 목표에 부합하지 않습니다. 따라서 대체 데이터셋을 찾기 위해 검색을 수행한 결과, Kaggle 웹사이트에서 제공되는 '과일 및 채소 이미지 인식 데이터셋'이 프로젝트의 목표와 더 관

련이 있는 것으로 확인되었습니다. 하지만 재료의 범위가 과일과 채소로만 제한되어 있습니다.

선택한 데이터 세트에는 그림 7에서 볼 수 있듯이 유사한 이미지가 있는 세 개의 서로 다른 폴더가 포함되어 있습니다. 이 기능을 사용하면 다양한 방법으로 모델을 구현할 수 있으며, 구현된 코드 이미지와 함께 나중에 자세히 설명하겠습니다. 이 분할 작업은 코드 블록을 통해 수행되므로 일반적으로 데이터 세트는 분할되지 않는다는 점에 유의할 필요가 있습니다. 그러나 이 프로젝트에 사용된 데이터 세트는 이미 분할되어 있으므로 코드에서 이 분할을 수행할 필요가 없습니다.

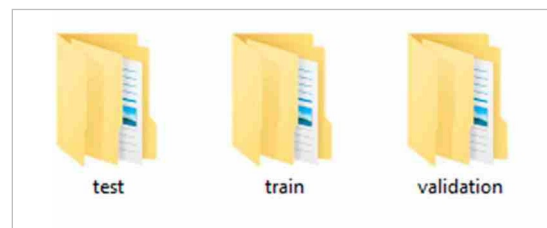


그림 7. 데이터 집합 폴더.

훈련 폴더에는 데이터 세트의 모든 이미지, 즉 모델이 프로세스 중에 훈련하고 학습하는 데 사용할 이미지가 들어 있습니다. 유효성 검사 폴더는 모델을 평가하는 데 사용되며, 이를 통해 모델 하이퍼파라미터를 조정하면서 훈련 데이터세트에서 모델을 편향 없이 평가할 수 있습니다. 유효성 검사 폴더에 있는 이미지를 통해 얻은 결과가 사용되며, 이를 통해 하이퍼파라미터가 증가하여 모델에 간접적으로 영향을 미칩니다. 주요 목적은 모델이 과적합하지 않은지 확인하는 것입니다(과적합은 모델이 이전에 관찰된 특정 데이터 세트에 매우 잘 맞지만 새로운 결과를 예측하는 데 효과적이지 않은 경우로 설명됩니다).

테스트 폴더의 기능은 훈련 폴더에 있는 이미지 세트를 기반으로 최종 모델을 평가하여 최종 모델의 정확도 수준을 결정하는 것입니다. 위에서 언급한 각 폴더에는 동일한 수의 클래스, 즉 36개의 식재료 클래스가 있으며, 각 클래스 안에는 그림 8과 같이 해당 식재료의 이미지가 있습니다.



그림 8. 데이터 집합 클래스의 폴더.

Kaggle에서 제공되는 "과일 및 채소 이미지 인식 데이터 세트"에 포함된 이미지의 해상도는 데이터 세트의 특정 버전과 이미지의 출처에 따라 다양합니다. 그러나 제시된 데이터 세트에는 해상도가 다른 이미지 세트가 포함되어 있습니다.

---

예시: 이 예에서는 데이터 세트의 이미지 중 하나가 그림 9와 같이 train 폴더에 있는 것으로 확인되며 크기는  $2000 \times 1000$ 입니다.

```

from PIL import Image

# Path to the directory containing the dataset images
dataset_dir = "C:/Users/rodri/Desktop/Dataset/train/banana"

# Load an example image from the dataset
image_path = dataset_dir + "/Image_27.jpg"
image = Image.open(image_path)

# Get the resolution of the image
image_width, image_height = image.size
print("Image resolution:", image_width, "x", image_height)

Image resolution: 2000 x 1000

```

그림 9. 이미지 해상도 예시.

따라서 사용된 데이터 세트의 특성화입니다:

- 36가지 종류의 식재료;
- 총 3115개의 이미지가 있습니다;
- 과일과 채소 이미지;
- PNG 이미지 형식의 이미지;
- 다양한 크기와 해상도;
- 다양한 각도, 광도 유형 및 품질의 이미지를 제공합니다.

### 3.3. 결과 및 토론

이 프로젝트의 모델 구축은 Anaconda 소프트웨어[27], 노트북 주피터[28]에서 사용할 수 있는 도구로 만든 ResNet-50 모델을 기반으로 했으며, 일련의 반복을 통해 더 나은 정확도를 얻기 위해 최상의 레이어와 활성화 기능을 선택하여 연속적으로 조정했습니다. 컴퓨터, 소프트웨어 및 라이브러리의 사양은 다음과 같습니다: 컴퓨터: Windows 10 Home, 버전 22H2, 인텔 코어 i7.4510U, 8GB RAM, 64비트 운영 체제; 사용된 소프트웨어 도구: 아난콘다 네비게이터 2.4.1, 주피터 노트북 6.5.4, 스파이더 5.3.3; 사용된 라이브러리: Matplotlib 3.4.3, Numpy 1.21.5, Pandas 1.3.4. 앞서 언급했듯이 총 3115개의 식재료 이미지가 포함된 '과일과 채소' 데이터셋을 선택했으며, 이 데이터셋 이미지로 CNN 모델을 훈련하고 인식된 식재료의 카테고리를 예측하는 것을 목표로 합니다. 먼저 노트북 주피터에서 새 파일을 생성한 후, 모델 구현에 필요한 케라스, 텐서플로우, 판다스, 넘피, 매트플롯리브 등의 라이브러리를 포함하는 코드 블록을 생성합니다.

다음으로, 데이터 증강(데이터 증강은 기존 데이터에서 약간 수정된 복사본을 추가하여 주어진 데이터 세트의 데이터 다양성을 증가시킴으로써 데이터의 양을 인위적으로 증강하는 일련의 기술[29]로, 그림 10과 같이 데이터 세트 이미지의 회전 및 방향 수정을 수행하여 CNN 모델이 동일한 이미지의 학습 과정을 다른 위치에서 수행하도록 하는 것을 목표로 구현했습니다.



그림 10. 이미지의 데이터 증강.

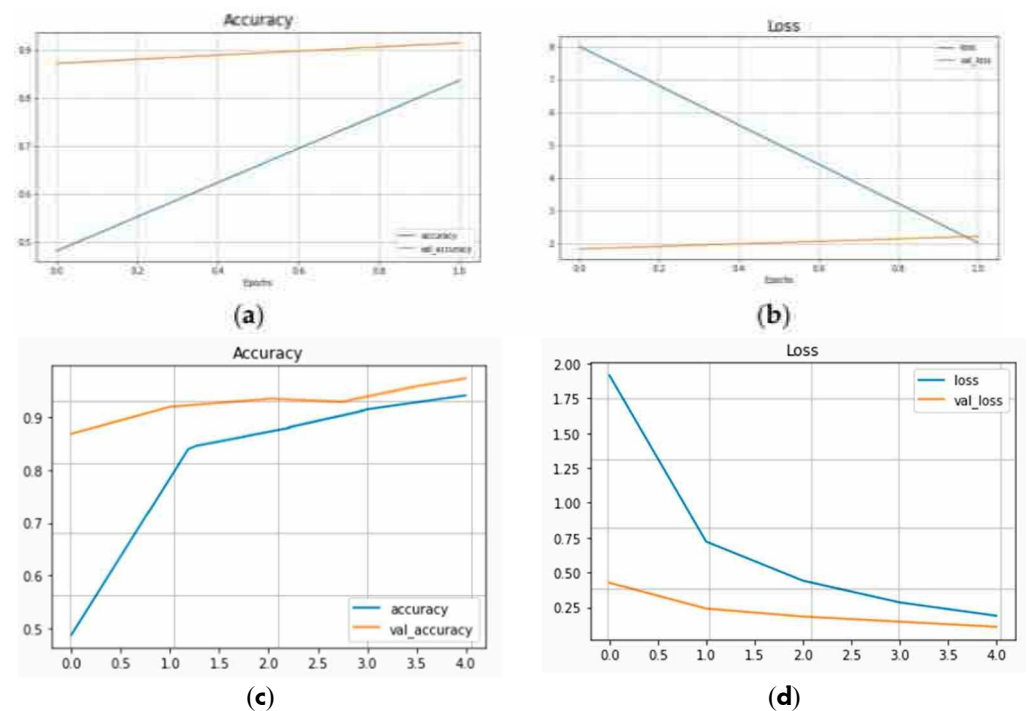
마지막 단계에서 ResNet-50 컨볼루션 신경망 모델은 평탄화 레이어, ReLu 함수가 포함된 활성화 레이어, 밀도 레이어, 소프트맥스 함수가 포함된 활성화 레이어 등 4개의 레이어를 추가하여 수정되었습니다. 이러한 추가 레이어는 CNN 모델의 정확도를 향상시키기 위해 추가되었습니다.

이러한 수정을 통해 모델을 생성한 후 정확도를 검증하기 위해 학습을 수행했으며, 초기 테스트에는 4개의 에포크(에포크는 학습 알고리즘이 전체 학습 데이터 세트를 통과하는 횟수를 정의하는 하이퍼파라미터)만 정의했습니다. 이 에포크 수가 상대적으로 적다는 점에 유의하는 것이 중요합니다.

이 단계의 목적은 모델의 기본 기능을 테스트하는 것입니다. 필요한 에포크 수는 데이터 집합에 따라 달라질 수 있으며, 데이터 집합이 클수록 더 많은 에포크가 필요합니다.

백분율로 측정되는 정확도 값은 학습 후 모델이 올바르게 분류한 이미지의 비율을 나타냅니다. 오류 값은 이미지를 분류할 때 모델의 정확도 수준을 나타냅니다.

Matplotlib 라이브러리를 사용하여 에포크에 따른 적중률의 정밀도를 시각화했습니다. 그림 11a는 에포크 수가 증가함에 따라 적중 값이 증가하는 모델의 정확도를 보여 주며, 훈련 데이터 세트(정확도)에서 약 92%, 테스트 데이터 세트(정확도)에서 83%의 정확도를 달성했습니다(val\_accuracy). 그림 11b는 이미지를 분류하는 모델의 동작을 보여주며, 에포크에 따라 손실 값이 감소하는 것을 볼 수 있습니다. 훈련 손실(loss)이 검증 손실(val\_loss)보다 낮다는 점에 주목할 필요가 있습니다. 또한, [그림 11c]는 4개의 에포크를 사용한 정확도를, [그림 11d]는 4개의 에포크를 사용한 손실을 표시한 것입니다.



**그림 11.** (a) 첫 번째 테스트의 정확도 그래프, (b) 첫 번째 테스트의 손실 그래프, (c) 4개의 에포크가 있는 테스트의 정확도 그래프, (d) 4개의 에포크가 있는 테스트의 손실 그래프.

모델의 적중률에 대한 최종 값은 여러 에포크와 이전 값으로 모델을 테스트하여 생성되었습니다. 전체 적중률은 두 개의 에포크 후에 약 90%에 도달했으며, 이는 이 초기 단계에서 모델의 적중률이 매우 만족스럽다는 것을 나타냅니다. 또한 이 단계에서 모델의 정확도도 만족스러운 수준입니다.

결론적으로, 특정 이미지에 대한 모델의 효과와 정확성을 검증하기 위해 Jupyter Notebook 도구 내에서 테스트를 수행했습니다. 10개의 이미지를 선택하고 모델의 예측을 각 이미지의 실제 클래스와 비교하기 위해 작은 코드를 만들었습니다. 각 이미지에 캡션을 추가하여 예측과 해당 클래스를 표시했습니다. 그림 12에서 볼 수 있듯이 모델은 모

---

은 이미지를 정확하게 분류하여 해당 클래스를 인식하는 데 있어 높은 정확도를 보여주었습니다.



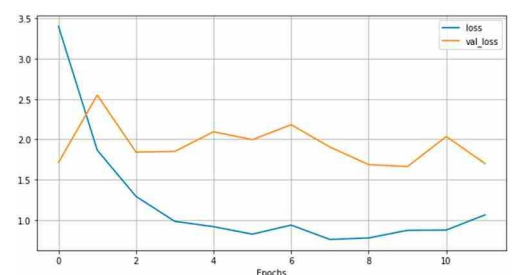
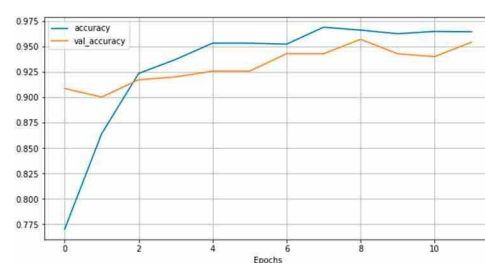


그림 12. 모델 정확도 검증.

그 후 결과를 개선하기 위해 모델 구현을 한 번만 변경했습니다. 첫 번째 테스트에서 얻은 값과 비교하여 모델이 더 정확한 이미지 분류를 수행할 수 있는지 확인하기 위해 모델 학습에 대한 에포크 수를 12개로 늘린 것입니다.

3115개의 이미지만 포함된 데이터 세트의 제한된 크기로 인해 에포크 수를 훨씬 더 늘릴 수 없다는 점에 유의해야 합니다. 매우 많은 수의 에포크를 사용하면 모델에 과적합이 발생하여 최종 값이 반복적으로 부정확해져 모델 성능에 부정적인 영향을 미칠 수 있습니다.

이 최종 테스트에서 정답 값은 점차 증가하여 7회차부터 일정하게 유지되어 약 96%의 적중률을 기록했습니다. 검증 값(val\_accuracy)의 경우, 그림 13a에서 볼 수 있듯이 점차 증가하여 약 95%의 값에 도달했습니다. 그림 13b의 그래프를 보면 테스트 값(loss)과 검증 값(val\_loss) 모두에서 손실 값이 감소하고 있음을 확인할 수 있습니다. 마지막 에포크에서는 테스트 값은 약간 증가하고 검증 값은 크게 감소했습니다. 훈련 손실에 대해 얻은 값은 에포크 7 이후 증가합니다. 3.2절에서 이미 언급한 이유 중 하나는 과적합으로, 이는 모델이 학습 데이터의 패턴을 포착하는 데 너무 특화되어 특정 조건, 특히 데이터 세트가 작을 때 일반화할 수 없음을 의미합니다. 데이터 세트가 작을수록 모델이 학습 데이터의 패턴을 학습하는 데 필요한 에포크 수 또한 줄어듭니다. 즉, 에포크 수를 늘리면 결국 모델이 과적합하게 됩니다. 이 경우 정확히 이런 일이 발생할 수 있습니다. 에포크 7에서 모델은 이미 모든 패턴을 인식했습니다. 이 경우에는 큰 문제가 아니며 모델 레이어를 조정하거나 더 큰 데이터 세트를 사용하여 해결할 수 있습니다.



(a) (b)

그림 13. (a) 두 번째 테스트의 정확도 플롯, (b) 두 번째 테스트의 손실 그래프.



이 이미지를 앞서 개발한 인식 모델로 분석하여 인식된 식재료가 사용된 레시피의 전체 재료와 조리법을 시각화할 수 있도록 했습니다. 이 솔루션의 아키텍처는 그림 15에 도식화되어 있습니다.

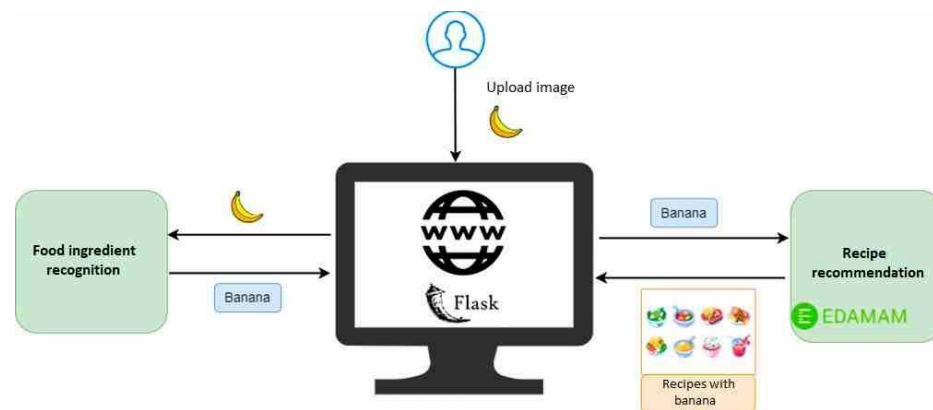


그림 15. 프로젝트 워크플로.

이 솔루션을 만들기 위해 스파이더 소프트웨어를 통해 플라스크 프레임워크를 사용했으며, 레시피를 추천하기 위해 에다맘[30]이라는 API를 사용했습니다.

#### 4.2. 이미지 인식

이 식재료 인식 컴포넌트의 목표는 사용자가 원하는 식재료를 인식하여 어떤 식재료인지, 그리고 나중에 인식된 식재료가 포함된 레시피를 식별할 수 있도록 하는 것입니다. 재료를 인식하는 방법에는 영상 인식, 재료에 대한 음성 입력 인식, 재료 이미지 인식 등 여러 가지가 있습니다. 이 논문에서는 사용자가 제공한 이미지나 사진을 이용한 식재료 인식의 가능성에 대해 살펴봅니다.

그림 15에서 사용자는 웹 애플리케이션 홈페이지의 업로드 옵션을 통해 이미지를 입력합니다. 다음으로 구현된 모델인 ResNet-50을 통해 성분을 인식합니다.

음식 재료를 인식하는 작업을 수행하기 위해 플라스크 마이크로 프레임워크가 활용되었습니다. Flask는 웹 애플리케이션을 생성할 수 있는 프레임워크로, CNN 모델을 구현하는 데 널리 사용됩니다. 따라서 Flask 프레임워크를 사용하여 두 개의 경로를 설정하여 CNN 모델을 구현하는 웹 애플리케이션을 개발했습니다:

- 경로 1: 그림 16과 같이 사용자가 식재료의 이미지를 업로드할 수 있는 인덱스 페이지입니다.
- 경로 2: 식재료 이미지를 인식하고 생성된 모델로 전송하여 식재료 등급을 결정하는 예측 페이지입니다. 그런 다음 이 정보는 그림 17과 같이 웹 페이지로 반환됩니다.

#### 4.3. 레시피 추천 시스템

사용자가 선택한 식재료를 기반으로 레시피를 추천하기 위해 자바스크립트 언어를 사용하여 인식된 식재료로 데이터베이스인 API Edamam을 검색하여 인식된 식재료가 포함된 레시피를 반환합니다.

또한 그림 17과 같이 사용자가 추천 레시피에서 식단이나 선호도를 선택할 수 있는 몇 가지 옵션도 만들었습니다. 사용자가 선택한 옵션에서 프로그램은 데이터베이스에서

검색을 수행하여 인식된 성분의 결과를 사용자가 선택한 식단 옵션 및 선호도와 함께 반환합니다. 사용자가 식단 양식에서 선택할 수 있는 옵션은 고단백, 저지방, 저탄수화물, 고섬유질, 균형 잡힌 식단입니다. 그리고 사용 가능한 기본 설정은 다음과 같습니다: 없음, 설탕 감소, 알코올 없음, 땅콩 무함유, 견과류 무함유, 채식주의자.

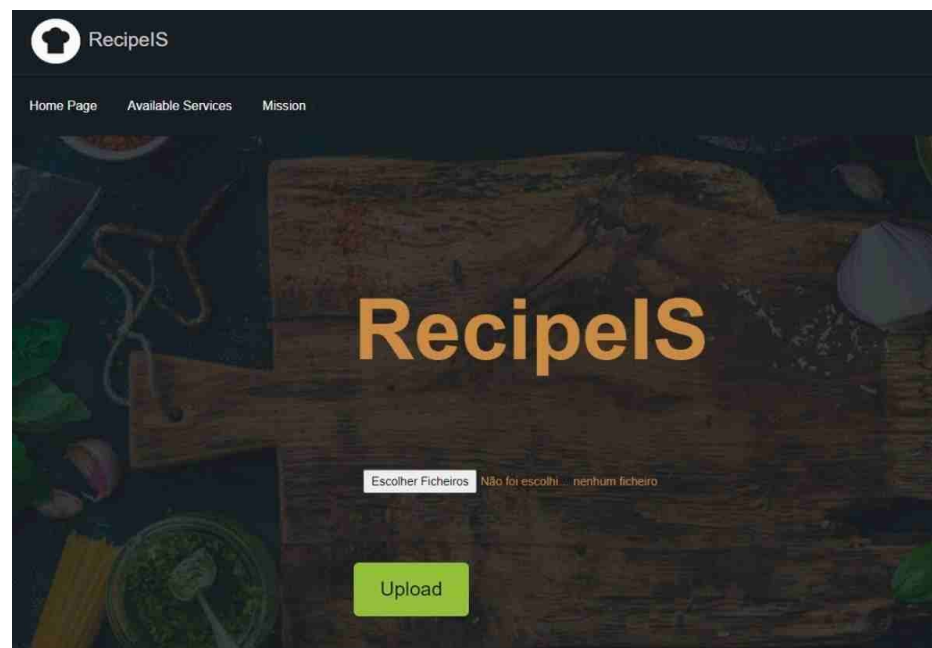



그림 16. 사용자를 위한 이미지 업로드 옵션이 있는 색인 페이지 인쇄.

Recognized Image



The recognized ingredient is cabbage!

Please select main ingredient: cabbage

Please select diet type: High Protein

Please select desired preference: Alcohol-Free

Submit

Your Search Results:

그림 17. 식단 및 선호도와 함께 인식된 성분의 인쇄물.

마지막으로 그림 18과 같이 레시피를 만드는 데 필요한 모든 재료와 사용자가 선택한 레시피의 웹사이트에 액세스하여 특정 레시피에 대한 지침과 자세한 정보를 얻을 수 있는 링크가 포함된 추천 레시피가 반환됩니다.



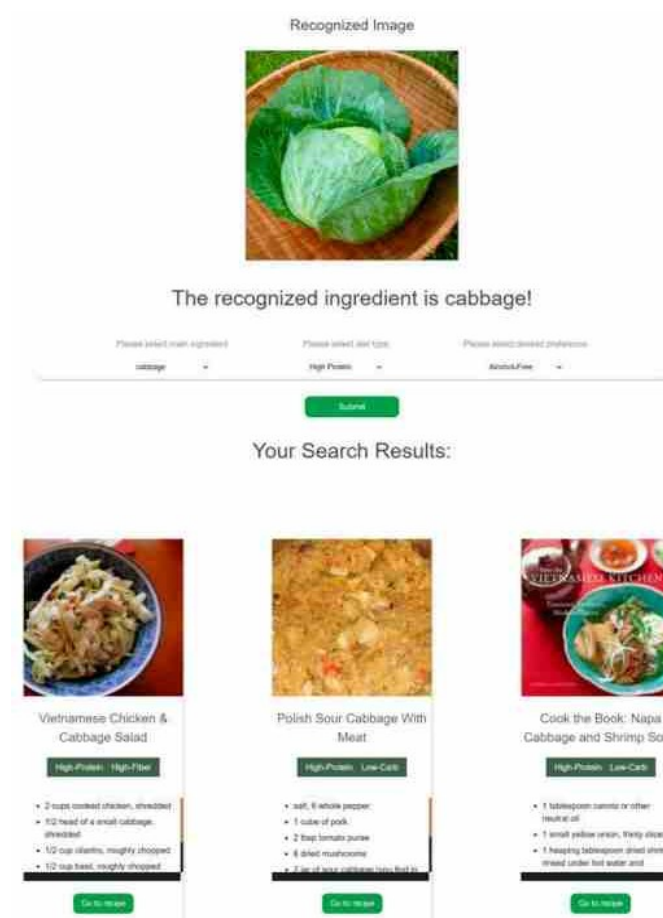


그림 18. 레시피 추천이 포함된 예상 웹 페이지 인쇄(이 예에서는 양배추를 주재료로 하는 레시피).

## 5. 결론

본 논문에서는 식재료 인식을 위한 컨볼루션 신경망 모델과 개념 증명으로 인식된 식재료를 기반으로 레시피를 추천하는 방법을 제시합니다. 제안한 모델은 마지막 부분에 레이어가 추가된 ResNet-50이며, 36개의 식재료 클래스가 포함된 데이터셋이 사용되었습니다. 이 데이터 세트로 모델을 학습시킨 후 3.2절에서 언급했듯이 두 번의 테스트에서 서로 다른 결과를 얻었습니다. 첫 번째 테스트에서는 90%의 정확도를 달성했고, 두 번째 테스트에서는 96%의 정확도를 달성하여 프로젝트 개발에 적합한 모델임을 입증했습니다.

추천 작업에서는 데이터베이스를 만들지 않고도 레시피 추천이 가능했기 때문에 Edamam API를 사용한 것이 중요한 선택이었습니다. 또한 필터를 포함할 수 있어 사용자에게 보다 개인화된 선택을 제공할 수 있었습니다.

향후 작업에서는 더 많은 음식 이미지와 클래스가 포함된 더 큰 훈련 데이터 세트를 사용하여 더 많은 레시피 세트에 액세스하는 것이 중요할 것입니다. 보다 상용화된 솔루션의 또 다른 기능은 사용자가 여러 개의 재료 이미지를 업로드할 수 있도록 하여 프로그램이 여러 재료를 동시에 인식하고 조합된 재료로 레시피를 추천할 수 있도록 하는 것입니다. 또한 재료를 추가하거나 제거하여 보다 구체적인 레시피를 추천하는 것도 흥미로울 것입니다. 같은 용도의 모바일 애플리케이션을 만들면 사용자가 실시간으로 식재료



---

이미지를 인식하고 레시피를 추천받을 수 있어 유용할 것입니다.

이 프로젝트는 사용자 지정 데이터베이스를 처음부터 개발하는 대신 Edamam API를 데이터베이스로 사용했다는 점에 주목할 필요가 있습니다. 이 결정은 광범위한 프로그래밍의 필요성을 없앴을 뿐만 아니라 레시피 추천에 대한 새로운 접근 방식을 도입했습니다. 철저한 연구와 조사를 통해 다음을 활용하면 다음과 같은 이점을 얻을 수 있다는 사실을 발견했습니다.

Edamam API의 기능은 레시피 추천을 생성하는 데 유망하고 효율적인 방법임이 입증되었습니다. 하지만 향후 작업으로 추천 시스템을 프로젝트에 통합하여 사용자 경험을 개선하고 전반적으로 완성도를 높이는 방안을 모색하는 것이 좋습니다. 검색할 수 있는 추천 시스템의 몇 가지 예로는 유사한 사용자의 선호도를 활용하여 통찰력 있는 추천을 제공하는 협업 필터링 기반 시스템이 있습니다. 이러한 조사와 구현을 통해 보다 효과적인 추천 시스템을 만드는 것이 목표입니다.

**저자 기여:** 개념화, 방법론, 석사, 박사, 박사과정, 소프트웨어, 검증, 조사, 리소스, 데이터 큐레이션, 초안 작성, 석사, 박사, 초안 검토 및 편집, 석사, 박사, 박사과정, 감독, 박사, 박사과정 모든 저자는 출판된 버전의 원고를 읽고 이에 동의했습니다.

**자금 지원:** 이 연구는 외부 자금 지원을 받지 않았습니다. **기관**

**검토 위원회 성명서:** 해당 없음. **정보에 입각한 동의서:** 해당 없음.

음.

**데이터 가용성 정책:** 해당 없음.

**이해 상충:** 저자는 이해 상충이 없음을 선언합니다.

## 참조

1. 텐서플로우. 사용 가능 온라인: <https://www.tensorflow.org/> (2022년 10월 26일에 액세스).
2. Keras: 파이썬 딥러닝 API. 온라인에서 사용 가능: <https://keras.io/> (2022년 10월 26일에 액세스).
3. NumPy. 온라인에서 사용 가능: <https://numpy.org/> (2022년 10월 26일에 액세스).
4. Page, M.J.; Moher, D.; Bossuyt, P.M.; Boutron, I.; Hoffmann, T.C.; Mulrow, C.D.; Shamseer, L.; Tetzlaff, J.M.; Akl, E.A.; Brennan, S.E. 외. PRISMA 2020 설명 및 상세화: 체계적 문헌고찰 보고를 위한 업데이트된 지침 및 예시. *BMJ* **2021**, *372*, n160. [CrossRef] [PubMed]
5. IEEE Xplore. 온라인에서 사용 가능: <https://ieeexplore.ieee.org/Xplore/home.jsp> (2022년 10월 26일에 액세스).
6. ACM 디지털 라이브러리. 온라인 이용 가능: <https://dl.acm.org/> (2022년 10월 26일에 액세스).
7. ScienceDirect.com|과학, 건강 및 의학 저널, 전문 기사 및 서적. 온라인 이용 가능: <https://www.sciencedirect.com/> (2022년 10월 26일에 액세스).
8. 유누스, R.; 아리프, O.; 아프잘, H.; 암자드, M.F.; 압바스, H.; 보카리, H.N.; 하يدر, S.T.; 자파르, N.; 나와즈, R. 딥러닝 기법을 사용하여 식품의 영양가를 실시간으로 추정하는 프레임워크. *IEEE Access* **2019**, *7*, 2643-2652. [CrossRef]
9. 첸, J.; 주, B.; 응, C.W.; 추아, T.S.; 장, Y.G. 식품 성분을 위한 다중 작업 및 지역 현명한 딥 러닝 연구 인식. *IEEE Trans. 오/이미지 프로세스* **2021**, *30*, 1514-1526. [CrossRef] [PubMed]
10. Pan, L.; Pouyanfar, S.; Chen, H.; Qin, J.; Chen, S.C. DeepFood: 딥러닝을 이용한 식품 성분의 자동 다중 클래스 분류. 2017 IEEE 제3회 협업 및 인터넷 컴퓨팅 국제 컨퍼런스, CIC 2017, 미국 캘리포니아주 새너제이, 2017년 10월 15-17일; 181-189쪽. [CrossRef]
11. 추, W.T.; 린, J.H. 딥 기반 공동 식품 카테고리, 재료 및 조리 방법 인식에 기반한 음식 이미지 설명. 2017 IEEE 국제 멀티미디어 및 엑스포 워크숍, ICMEW 2017, 홍콩, 중국, , 2017년 7월 10-14일; 109-114쪽. [CrossRef]
12. Zhang, L.; Zhao, J.; Li, S.; Shi, B.; Duan, L.Y. 마켓에서 요리까지: 개인화된 레시피 추천을 위한 다중 재료 이미지 인식. 2019 IEEE 국제 멀티미디어 및 엑스포 컨퍼런스(ICME); IEEE: 미국 뉴욕 맨해튼, 2019;

- pp. 1252-1257. [[CrossRef](#)]
13. 살바도르, A.; 드로즈잘, M.; 지로-이니에토, X.; 로메로, A. 역조리: 음식 이미지에서 레시피 생성. 2019년 6월 15~20일, 미국 캘리포니아주 롱비치에서 열린 컴퓨터 비전 및 패턴 인식에 관한 IEEE 컴퓨터 학회 컨퍼런스 논문집; pp. 10445-10454. [[CrossRef](#)]
  14. 왕, H.; 사후, D.; 리우, C.; 임, E.P.; 호이, S.C.H. 요리 레시피와 음식 이미지를 위한 적대적 네트워크를 이용한 교차 모달 임베딩 학습. 컴퓨터 비전 및 패턴 인식에 관한 IEEE 컴퓨터 학회 컨퍼런스 논문집, , 미국 캘리포니아주 롱비치, 2019년 6월 15-20일; 11564-11573쪽. [[CrossRef](#)]
  15. 음식 인식을 위한 다중 스케일 다중 뷰 심층 기능 집계. *IEEE Trans. 이미지 프로세스*. **2020**, *29*, 265-276. [[CrossRef](#)]
  16. Liang, H.; Wen, G.; Hu, Y.; Luo, M.; Yang, P.; Xu, Y. MVANet: 중국어를 위한 멀티태스크 가이드 멀티뷰 주의 네트워크 음식 인식. *IEEE Trans. Multimed.* **2021**, *23*, 3551-3561. [[CrossRef](#)]
  17. Chen, J.; Ngo, C.W. 요리 레시피 검색을 위한 딥 기반 재료 인식. 제24회 ACM 멀티미디어 국제 학술대회 논문집, 네덜란드 암스테르담, 2016년 10월 15-19일; 32-41쪽. [[CrossRef](#)]

18. An, Y.; Cao, Y.; Chen, J.; Ngo, C.W.; Jia, J.; Luan, H.; Chua, T.S. PIC2DISH: 맞춤형 요리 어시스턴트 시스템. 2017년 10월 23-27일, 미국 캘리포니아주 마운틴뷰에서 열린 제25회 ACM 멀티미디어 국제 학술대회 논문집; 1269-1273쪽. [CrossRef]
19. 임, H.; 황, X.; 밀러, S.; 에델만, J.; 유켄, T.; 보이다, S. 스마트 쿡: 멀티 모달 학습으로 더 쉬운 요리 만들기. 2019 퍼베이시브 및 유비쿼터스 컴퓨팅에 관한 2019 ACM 국제 공동 컨퍼런스 및 2019 웨어러블 컴퓨터에 관한 2019 ACM 국제 심포지엄의 부속 논문집, 영국 런던, 2019년 9월 9-13일; pp. 129-132. [CrossRef]
20. Zhu, B.; Ngo, C.W.; Chen, J.J. 도메인 간 교차 모드 식품 전송. 제28회 ACM 국제 학술대회 멀티미디어 세션( ), 미국 워싱턴주 시애틀, 2020년 10월 12-16일; 3762-3770쪽. [CrossRef]
21. 모바일 e-헬스 멀티미디어 애플리케이션을 위한 지능형 클라우드 기반 데이터 처리 브로커. *Future Gener. Comput. Syst.* **2017**, *66*, 71-86. [CrossRef]
22. 카가야, H.; 아이자와, K.; 오가와, M. 컨볼루션 신경망을 이용한 음식 감지 및 인식. 2014년 11월 3-7일, 미국 플로리다주 올랜도, 제22회 ACM 멀티미디어 국제 컨퍼런스, 1085-1088쪽. [CrossRef]
23. Wang, W.; Duan, L.Y.; Jiang, H.; Jing, P.; Song, X.; Nie, L. Market2Dish: 건강을 고려한 식품 추천. *ACM Trans. Multimed. Comput. Commun. (TOMM)* **2021**, *17*, 33. [CrossRef]
24. 주석이 달린 ResNet-50. ResNet-50의 작동 방식 및 인기 이유 설명(Suvaditya Mukherjee 작성) | 데이터 과학을 향하여. 온라인에서 사용 가능: <https://towardsdatascience.com/the-annotated-resnet-50-a6c536034758> (에서 액세스됨, 2022년 10월 31일).
25. 잔여 네트워크(ResNets)의 이해 및 구현 | 작성자: Raghunandepu | 분석 Vidhya | 매체. 온라인에서 사용 가능: <https://medium.com/analytics-vidhya/understanding-and-implementation-of-residual-networks-resnets-b80f9a507b9c> (2022년 10월 31일에 액세스).
26. 딥러닝에서 연결 건너뛰기에 대한 직관적인 설명|AI Summer. 온라인에서 사용 가능: <https://theaisummer.com/skip-connections/> (2022년 10월 31일에 액세스).
27. 아나콘다란 무엇인가요? 도미노 데이터 과학 사전. 온라인에서 사용 가능: <https://www.dominodatalab.com/data-science-사전/아나콘다>(2023년 4월 27일에 액세스).
28. 프로젝트 주피터|홈. 온라인 이용 가능: <https://jupyter.org/> (2023년 4월 27일에 액세스).
29. 데이터 증강이란 무엇인가요? 2023년 기술 및 사례. 온라인에서 사용 가능: <https://research.aimultiple.com/data-augmentation/> (2023년 4월 27일에 액세스).
30. 에다맘-푸드 데이터베이스 API, 영양 API 및 레시피 API. 온라인에서 사용 가능: <https://www.edamam.com/> (2022년 10월 26일에 액세스).

**면책 조항/발행인 참고 사항:** 모든 출판물에 포함된 진술, 의견 및 데이터는 전적으로 개별 저자 및 기고자의 것이며, MDPI 및/또는 편집자의 것이 아닙니다. MDPI 및/또는 편집자는 콘텐츠에 언급된 아이디어, 방법, 지침 또는 제품으로 인해 발생한 인명 또는 재산상의 피해에 대해 책임을 지지 않습니다.