캡스톤 디자인 결과보고서

주제	유기동물 입양 기간 예측	
팀명	팀원명 (총 3명)	담당교수
집▪돌	이상훈(팀장), 김윤하, 안중현	황두성 교수님

과제 개요

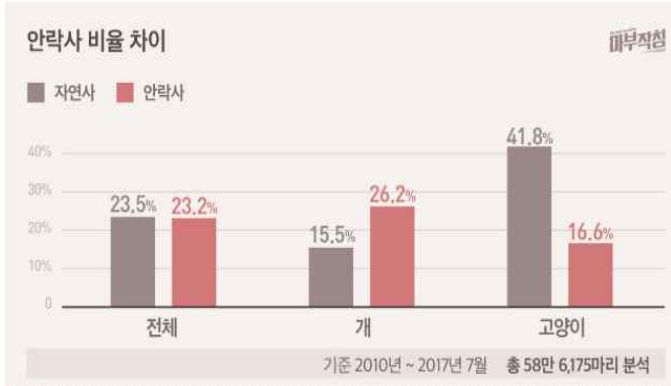
설계 과제 요약/목표

Deep Learning 과 Machine Learning 을 통해 메타데이터 및 이미지를 학습하여 동물 형태에 따른 입양기간을 예측한다. 그로 인해 유기동물보호소에서 학습된 모델을 이용하여 유기동물의 입양기간을 예측하고 입양기간이 길게 예측된 동물에 홍보를 집중하는 차등적 홍보등의 방법을 통한 유기동물의 입양이 순환이 잘 되도록 하여 더 많은 동물이 입양되어 안락사를 줄이고자 한다.

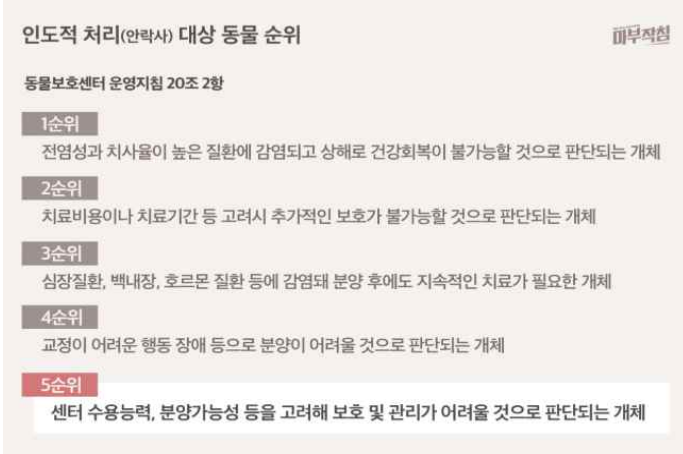
현황 분석

기존 방법의 특징/한계

▪ 수백만의 길 잃은 애완동물들이 유기되고 보호소로 보내진다. 유기동물들은 입양자를 기다리다 보호소 수용 최대가 되면 안락사를 당하게 된다.



-출처 한국동물구조관리협회



실제로 건강상에 문제가 있어서 동물들을 안락사하는 경우도 많지만 유기동물 보호센터의 수용능력에 문제와 효율적인 분양시스템이 이루어지지 않아서 안락사하는 경우도 많다.

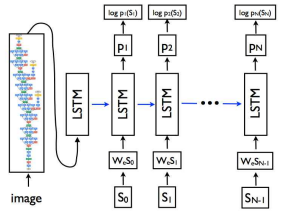
과제 내용

개발 프로세스 구현환경 진행일정

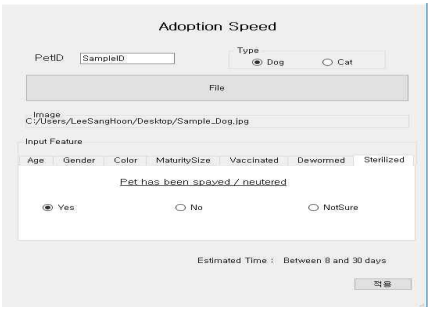
▪ Tensorflow, Keras : image 와 metadata 를 분석 및 학습하여 유기동물들의 입양기간을 예측하는 모델 제작
▪ PyQT : 학습이 완료된 머신러닝 모델을 적용하여 유기동물보호소에서 사용할 수 있는 간단한 GUI 모델 제작
▪ Pycharm, Anaconda : Python 개발환경, 머신러닝 개발환경
▪ Colab : 텍스트, 코드 출력을 하나의 공동작업 문서로 통합해 주는 머신러닝 교육과 연구를 위한 데이터 분석 도구
▪ Deep learning & Machine learing: 이미지 학습을 위한 imageNet 과 데이터 분류를 위한 Randomforest, K-NN, Rogistic Regression 활용

세부내용	수행기간(월)												비고
	3				4				5				
	1	2	3	4	1	2	3	4	1	2	3	4	
아이디어 기획 및 설계	■	■	■										
스터디 및 모델 탐색 데이터 분석		■	■	■	■								cs231/tensor
데이터 프로세싱				■	■	■	■						image, meta
모델 제작						■	■	■					vggNet, RF
Model Traning&Testing							■	■	■	■	■		
보고서 작성 및 제출											■	■	

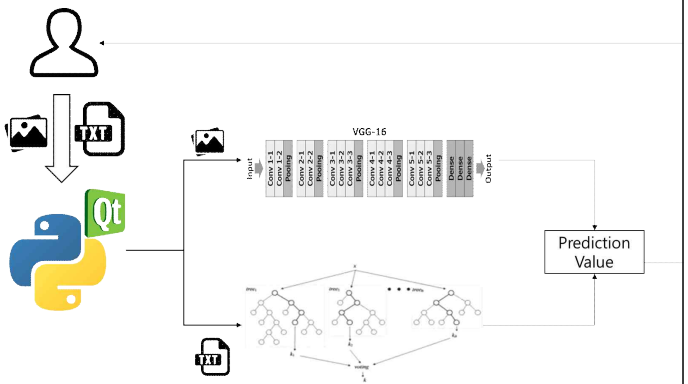
내용	수행 절차	1. Data Processing ▪ Metadata 중에서 학습에 방해가 되는 feature 들을 제외하였고 Image data 중에서는 동물 식별이 힘든 image 를 제거하였다. 2. Data Training ▪ 2-1 Metadata Machine Learning −KNN Algorithm −Random Forest Algorithm −Logistic Regression Algorithm 3가지의 알고리즘을 이용하여 입양기간 정확도가 가장 높은 Model을 선정 ▪ 2-2 Image Classification(Deep learning) CNN model(VGG, Resnet)을 이용해 image를 학습시켜 입양기간을 예측 3. GUI Program을 만들어 사용자가 data를 입력하여 모델을 통해서 예측된 입양기간을 출력해준다.
내용	프로젝트 진행 내용	개발은 Machine laerning 부분에서는 Random forest와 K-NN 모델들의 구현이 끝나고 Matadata들을 학습을 시켜보았고 Deeplearning CNN모델 중 Resnet을 구현하였고 VGG는 구현 중 생긴 오류를 수정하고 있었다. 완성된 모델들에 데이터를 학습시켜 보았는데 학습을 통한 예측의 정확도가 유의미한 결과가 나오지 못하였다. 추후 VGG와 logistic regression까지 완성이 된 후에 정확도를 측정하였지만 마찬가지로 높은 정확도를 보여주지는 못하였다. 데이터 전처리에 충분한 시간을 투자하였지만 더 이상 정확도가 올라가지는 않았다. 그래서 정확도가 낮은 다른 이유를 찾고자 하였다. 우선 ML 모델은 학습에서 오버피팅이 심하게 일어나 Depth를 줄이고 하이퍼 파라미터를 계속 조절하며 데이터 불균형을 줄이고자 하였다. 그 방법으로 오버 샘플링 방법을 찾았고 그 중 SMOTE 기법을 사용하였다. SMOTE 기법을 사용하여 ML 모델들의 정확도를 많이 끌어 올렸으며 특히 Random forest가 0.39 -> 0.49로 ML 모델들 중에 그나마 가장 높은 정확도를 보였다. DL CNN 모델에서 또한 정확도를 높이고자 하는 노력을 많이 하였는데, 그 중 개와 고양이를 따로 분리하여 학습을 시키는 것은 큰 효과가 없었다. 오히려 Activation이나 데이터 프로세싱 기법들과 같은 기본적인 변화들을 주었더니 VGG가 20%->37% 큰 폭으로 상승하였다. 분명 정확도는 많이 상승시켰지만 정형화되지 않은 데이터를 이용해서 인지

		실제로 사용하기는 힘든 유의미하지 않은 결과가 나왔다. 그렇기에 다른 방식을 찾아보았고 LSTM 기법을 도입해 보았다.  CNN 모델에서 사용하는 이미지 데이터와 ML에서 사용하던 Metadata를 엮어 Many-to-one 형식의 CNN-LSTM 구조를 만들었다. 이미지를 CNN에 통과시켜서 LSTM의 초기 데이터로 사용할 수 있도록 데이터를 편 다음 LSTM을 통화시키며 Metadata의 각 feature들을 적용하는 방식을 취하였으나 오히려 VGG와 Randomforest보다 정확도가 많이 낮게 나왔다. 그래서 다음 방법으로 이미지만 LSTM을 통과시키는 방법과 Meta data만 LSTM을 통과시키는 방법을 취해 보았지만 전과 큰 차이가 없어 결국 LSTM 도입은 실패하였다. 과제수행 기간이 막바지에 다다라 다른 방법을 찾고자 하다 단순하게 DL과 ML의 결과를 엮는 방식을 택하였다. 데이터들을 사용자가 입력할 시에 DL과 ML이 각각 결과를 낸 후 그 결과치를 종합하는 방식이었다. 우선 각 모델 선택은 DL과 ML에서 정확도가 가장 높은 모델로 Randomforest와 VGG를 선택하였다. 입양기간의 예측을 범위 단위로 진행하게 되는데 (0~7일 내 입양은 0, 7~21일은 1 등…) 두 모델이 입양기간을 다르게 예측할 경우 어떤 방식으로 범위를 선택해야 할지를 선택해야 했다. 그런데 입양기간 데이터 특성상 숫자가 클수록 점점 넓은 범위를 갖는 특성을 가지고 있기에 두 모델의 예측 기간 중 Max값을 가져오는 방법을 취하였다. 그리고 최종 정확도는 52%를 기록하였다. 처음 캡스톤의 결과물로 단순하게 Accuracy를 보여주는 형태를 취하려고 하였다. 그러나 멘토와의 만남에서 결과물이란 Visualize가 되어야 한다는 말씀을 해 주셨고 단순하게 숫자만 보여주는 것이 아닌 사람들이 보고 반응할 수 있는 결과물이 어떤 것이 있을 것인가를 같이 고민해 주셨다. 입양기간에 따라 우선순위가 자동으로 바뀌는 웹 어플리케이션 등 다양한 아이디어가 나왔으나 2차 멘토링을 진행하는 시기가 결과물 제출일까지 많은 시간이 남지 않았었고 그 때는 아직 정확도를 조정하는 단계여서 많은 시간을 투자할 수 없

었기에 PythonQT로 GUI를 만들어 사용자가 실습을 하고자 할 때 직접 Image와 Metadata 입력 시에 예측한 입양기간을 보여주는 Application을 만들었다.



Process를 도식 화하여 하단과 같이 나타내었다.

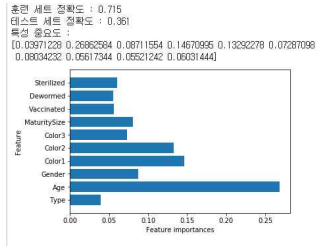


(Architecture)

ML과 DL 모델 결과 각각의 예측 결과를 단순 Max 취하는 것은 정확도가 오르더라도 오히려 설득력이 떨어지기에 실제로 쓰기에는 무리가 있다는 것을 알고 있다. 하지만 50%도 안되는 정확도로 프로젝트를 마무리하기 보다 다양한 모델들을 시도해 보았으며 정확도를 올리기 위해 다양한 방법을 사용해 보았다 라는 메시지를 전달하고 싶었다. 마지막 발표 때에 교수님께서 데이터 자체가 문제일 수도 있다 라는 말씀을 해 주셨다. 캡스톤 기간 동안은 유의미한 결과가 나오지 못했지만 데이터부터 다시 확인하며 유의미한 결과를 만들어내고자 한다.

결과 및 문제점

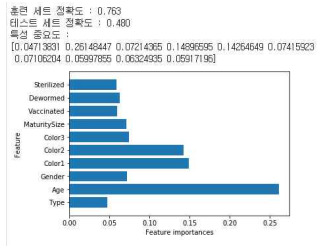
- Machine Learning model
  - 정확도: Random Forest > Logistic Regression > KNN



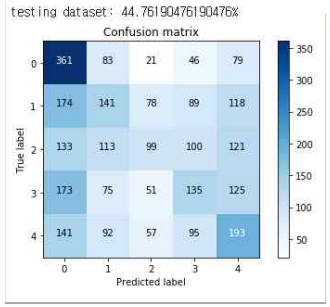
<Randomforest>

하지만 현실 데이터인 경우 불규칙 데이터가 많아 정확도가 낮게 나온다.
따라서 SMOTE(Synthetic Minority Over-Sampling Technique) 라는 Over-Sampling 기법을 사용하여 정확도를 올릴 수 있었다.

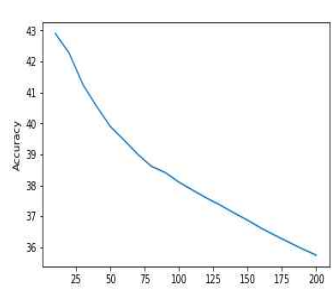
- SMOTE 사용 후 36% -> 48%



- 다른 Machine Learning Algorithm(K-NN, Logistic Regression) 들도 SMOTE기법 적용하여 다시 결과를 재비교



<Logistic Regression>

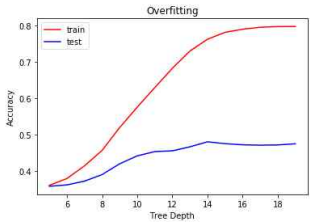


<K-NN>

결과 및 문제점

■ Overfitting

SMOTE를 통해서 Randomforest의 정확도가 올라갔지만 여전히 overfitting의 문제가 있었다.



(Random Forest의 overfitting)

따라서 parameter의 값들을 변경해본 결과 트리의 깊이 때문에 overfitting이 이루어짐을 알 수 있었다. 트리의 깊이를 얕게 하면 overfitting을 해결할 수 있었지만 동시에 정확도도 줄어들기에 parameter를 비교하여 최적의 parameter의 값을 설정하였다.

• Deep Learning model

VGG16 Model Architecture

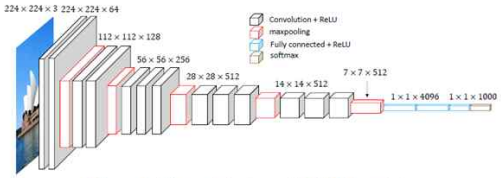
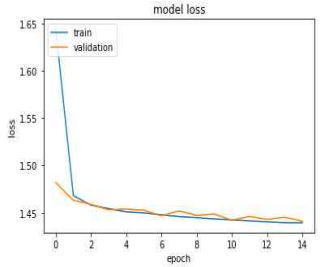


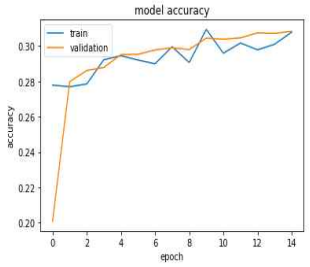
Figure 2: The architecture of VGG16 model .

VGGNet의 모델을 선택한 이유는 이해가 쉽고 변형을 시켜가면서 테스트하기에 용이하다는 것 이였다. 일단 layer층이 현재 다른 ResNet이나 GoogelNet들에 비해서 얕았기 때문에 단시간에 이해하기가 쉬웠던 것 같다.

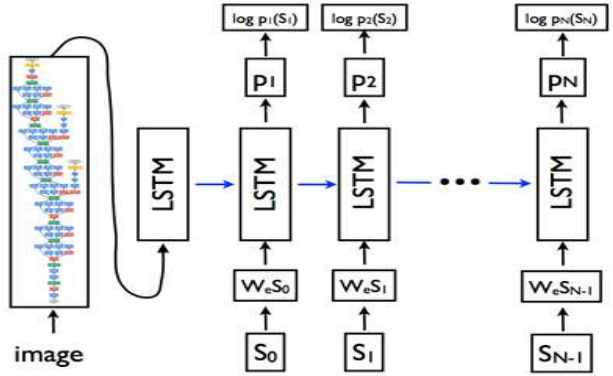
뿐만 아니라 VGGNet의 가장 큰 특징은 작은 filter 크기의 convolution 연산이기 때문에 그 만큼 parameter의 개수가 많아져서 연산량이 증가하여 학습시간이 오래 걸렸지만 높은 정확도를 얻을 수 있었다.

■ Image Training 결과





•LSTM&CNN



정확도 문제를 해결하기 위해서 CNN과 LSTM을 이용하여 이미지 데이터와 Metadata를 함께 사용하고자 하였고 이미지를 CNN을 통과시킨 후 LSTM을 이용하여 Many-one 예측 구조를 이용하려고 하였다. 하지만 오히려 다른 ML,DL 모델보다 성능이 더 떨어졌다.

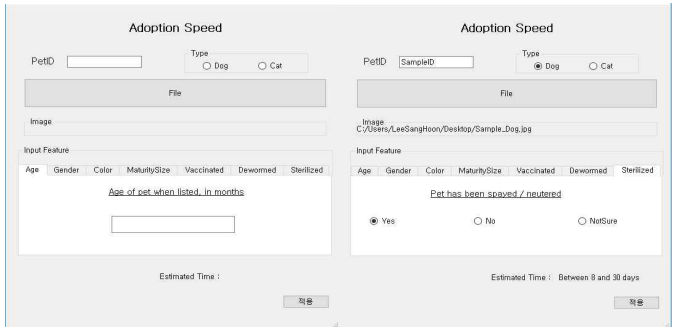
•결과 예측

각 모델들의 예측 값들과 실제 값이, 예측 값의 평균, 작은 값, 큰 값 중 어떤 값을 선택했을 때 정확도가 향상되는지 비교하였다.

	Prevoius acc	Last acc
VGG	20%	37%
Randomforest	36%	49%
VGG+Randomforest	-	52%

다른 두 방법은 떨어지거나 변화가 없는 데에 비해 높은 값으로 예측한 값을 선택하는 방법을 택했을 때에 정확도가 상승하였다. 그래서 VGG와 Randomforest 의 각 결과값 중 큰 값을 기준으로 택하는 방식을 취하였고 정답으로 예측한 모델 선정하여 최종 정확도는 52%를 기록하였다.

• GUI



QT designer Tool을 이용하여 Window의 틀을 만들고 PYQT5 Module을 이용, 각 button, text, label에 대한 event 처리하였고 model을 가져와 사용자들이 입력한 값들을 적용하여 입양기간의 예측값을 출력하였다.

결과물에 대한 기대 효과 및 활용 방안

□ 유기동물 입양 기간 예측
- ■ 유기된 동물의 입양 기간 예측 가능
- ■ 예측 입양기간에 따라 효율적 홍보 방식을 택하여 입양 순서의 원활한 순환구조를 만든다.
- ■ 예측에 따른 선별적 집중 홍보 가능
- ■ 선별적 집중에 따른 입양 실패 동물 감소
- ■ 입양 실패 감소에 따른 안락사 감소

□ 동물 보호소
- ■ 전체 동물 홍보에서 선별적 홍보방식으로 홍보 효율을 높임
- ■ 안락사 감소에 의한 동물보호단체의 이미지 상승