연세대학교 의과대학 주최, 보건산업진흥원 후원, 데이콘 주관

유방암의 임파선 전이 예측 AI 경진대회

임효정 (팀명 : lhj)
lim.gadi@gmail.com

TABLE OF CONTENTS

01 **EDA & Preprocessing**
Image Data Preprocessing, Tabular Data Preprocessing

02 **Modeling**
Multiple Instance Learning, CNN-Tabular Multi-modal Learning,
Tree Based Boosting Algorithm

03 **Cross Validation**
Stratified 5-Fold Validation

04 **Ensemble**
Final Model Architecture, Stacking

01 Preprocessing

Image Data Preprocessing

01 학습 방해요소 파악

02 Tissue Segmentation

03 Tissue 위주로 이미지 crop

04 Tile Creation

01 Preprocessing

병리 슬라이드 이미지에서의 학습 방해요소 파악

Image Data Preprocessing

지나치게 background
면적이 넓음

슬라이드 tissue 근처의 오염,
버블, 접힘 등 noise

슬라이드 가장자리 선 음영

01 Preprocessing

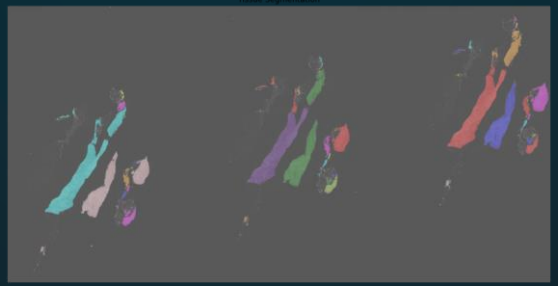
병리 슬라이드 이미지에서 Tissue Detection & Segmentation

Image Data Preprocessing

Image Segmentation 시에 grayscale로 Otsu threshold를 구하는 경우가 많지만, 이 병리 슬라이드 상의 오염이나 가장자리 음영 등 noise가 많아 grayscale로 threshold를 구하면 noise도 함께 인식되는 문제가 있다. 따라서 tissue가 붉은색 계열이라는 것에 착안하여 이미지를 LAB color space로 변환 후 A channel을 뽑아 Otsu threshold를 구하니 효과적으로 tissue 부분만 segmentation할 수 있었다.

원본 이미지

Tissue Segmentation 결과 시각화



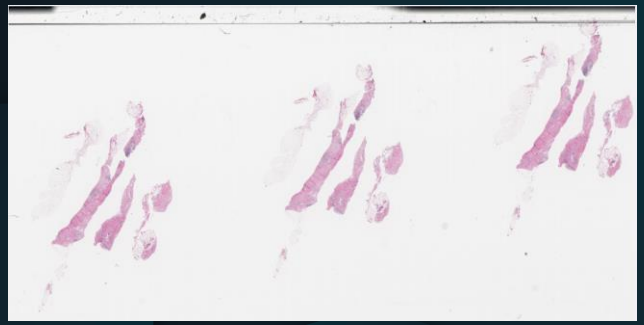
01 Preprocessing

Image Data Preprocessing

병리 슬라이드 이미지에서 Tissue Detection & Segmentation

Tissue Segmentation 후에 tissue가 아닌 부분에
white masking 처리를 진행하였다.

원본 이미지



Tissue Segmentation 및 masking 처리 이미지

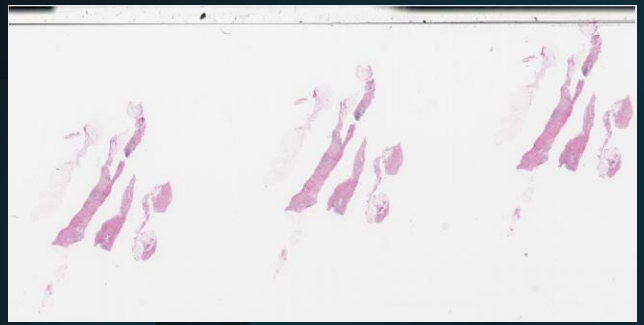
01 Preprocessing

Image Data Preprocessing

병리 슬라이드 이미지에서 불필요한 background 제거하여 tissue 위주로 crop

이미지를 행, 열 각각 for문으로 돌면서 전체 행 또는 전체 열이 white masking된 부분은 불필요한 부분이라는 전제 하에 remove한다. 이렇게 처리하면 tissue 부분 위주로 crop한 이미지를 얻을 수 있다.

원본 이미지



Crop한 이미지

01 Preprocessing

병리 슬라이드 이미지에서의 학습 방해요소 해결

Image Data Preprocessing

지나치게 background
면적이 넓음

슬라이드 tissue 근처의 오염,
버블, 접힘 등 noise

슬라이드 가장자리 선 음영

01 Preprocessing

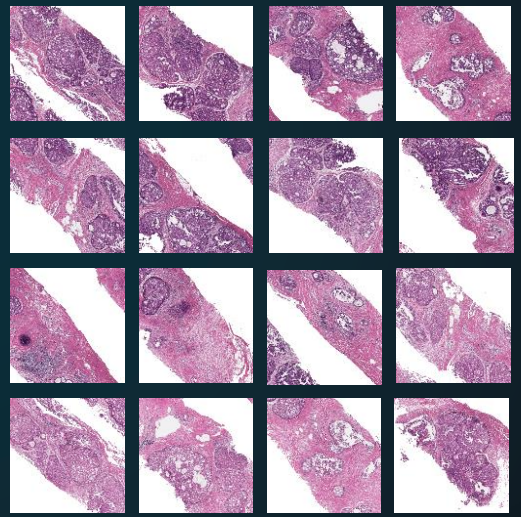
Image Data Preprocessing

Tissue 위주로 crop한 이미지를 바탕으로
Tile Creation

1. Tissue Segmentation 후 crop한 이미지에서 Tile Creation

2. 이미지의 height와 width 중 더 긴 쪽을 15등분하여 tile size 결정

3. Top 16 darkest tiles 선택



01 Preprocessing

Image Data Preprocessing

Tissue 위주로 crop한 이미지를 바탕으로 Tile Creation

What Works

1. **Tissue Segmentation 후 crop한 이미지에서 Tile Creation** : noise가 제거된 상태에서 안정적으로 tile을 생성 가능.
2. **Tile size를 상대적으로 이미지의 height와 width 중 더 긴 쪽을 n등분하여 정하기** : 이미지 데이터마다 크기가 상이하더라도 n등분을 하여 tile을 생성하면 전반적으로 균일한 스케일로 tile이 생성됨.
3. **Top 16 darkest tiles 선택** : tissue 위주로 만들어진 타일이 선택됨.

What Doesn't Work

1. **원본 이미지에서 Tile Creation** : noise가 많이 뽑힘. 특히 가장자리 음영, 불필요한 오염들 위주로 만들어지는 tile이 상당히 많았음.
2. **Tile size를 절대적인 픽셀값으로 정하기** : 이미지마다 크기가 상이하기 때문에 어떤 이미지는 지나치게 확대되어 tile이 생성되고 어떤 이미지는 지나치게 조직이 제대로 보이지 않는 크기로 tile이 생성되는 문제.
3. **Top 16 highest deviation tiles 선택** : deviation이 높을수록 tissue와 하얀색 background가 섞인 tile이 많이 선택되어서 기각.

01 Preprocessing

Tabular Data Preprocessing

01 EDA

02 결측치 처리

03 Feature Generation

01 Preprocessing

EDA 및 변수간 상관관계 파악

Tabular Data Preprocessing

변수간 피어슨 상관계수를 구하여 시각화

	나이	진단명	암의 위치	암의 개수	암의 장경	NG	HG	HG_score_1	HG_score_2	HG_score_3	DCIS_or_LCIS_여부	DCIS_or_LCIS_type	T_category	ER	ER_Allred_score	PR	PR_Allred_score	KI-67_LI_percent	HER2	HER2_IHC	HER2_SISH	HER2_SISH_ratio	BRCA_mutation	N_category
나이	1.00	-0.03	0.03	-0.11	0.13	0.04	0.03	0.08	0.03	-0.01	-0.02	0.08	0.10	-0.03	0.21	-0.18	0.04	-0.03	0.01	0.02	0.08	0.01	0.11	0.07
진단명	-0.03	1.00	-0.01	0.09	0.01	-0.18	-0.18	-0.27	-0.23	-0.14	0.05	-0.05	0.04	0.08	0.06	0.03	0.03	-0.10	-0.08	-0.02	-0.11	-0.07	-0.07	-0.05
암의 위치	0.03	-0.01	1.00	0.00	-0.01	0.06	0.04	0.02	0.03	0.05	0.05	-0.09	0.04	-0.02	0.02	-0.00	0.04	0.04	-0.02	-0.02	0.07	-0.09	0.12	-0.03
암의 개수	-0.11	0.09	0.00	1.00	0.01	0.09	0.03	0.09	0.06	-0.06	0.34	0.06	0.06	0.08	-0.02	0.04	0.03	0.04	-0.12	0.01	0.03	-0.07	0.04	0.27
암의 장경	0.13	0.01	-0.01	0.01	1.00	0.31	0.35	0.25	0.31	0.31	0.02	0.10	0.81	-0.14	0.08	-0.10	0.06	0.24	-0.00	-0.02	0.11	0.35	-0.08	0.29
NG	0.04	-0.18	0.06	0.09	0.31	1.00	0.76	0.45	0.98	0.51	0.09	0.26	0.38	-0.36	-0.05	-0.29	-0.02	0.52	0.15	0.17	0.24	0.13	0.05	0.37
HG	0.03	-0.18	0.04	0.03	0.35	0.76	1.00	0.65	0.76	0.73	0.08	0.31	0.31	-0.40	-0.07	-0.30	0.03	0.55	0.13	0.12	0.21	0.16	-0.01	0.25
HG_score_1	0.08	-0.27	0.02	0.09	0.25	0.45	0.65	1.00	0.43	0.28	0.14	0.31	0.23	-0.22	0.02	-0.16	0.02	0.30	0.04	0.03	0.22	0.09	-0.22	0.27
HG_score_2	0.03	-0.23	0.03	0.06	0.31	0.98	0.76	0.43	1.00	0.52	0.11	0.27	0.24	-0.36	-0.03	-0.27	0.00	0.51	0.12	0.15	0.23	0.13	0.04	0.29
HG_score_3	-0.01	-0.14	0.05	-0.06	0.31	0.51	0.73	0.28	0.52	1.00	-0.02	0.28	0.24	-0.40	-0.19	-0.27	-0.00	0.56	0.16	0.11	0.07	0.22	0.04	0.05
DCIS_or_LCIS_여부	-0.02	0.05	0.05	0.34	0.02	0.09	0.08	0.14	0.11	-0.02	1.00	0.38	-0.01	0.07	0.18	0.09	0.15	0.06	-0.19	0.07	0.05	-0.12	0.15	0.32
DCIS_or_LCIS_type	0.08	-0.05	-0.09	0.06	0.10	0.26	0.31	0.31	0.27	0.28	0.38	1.00	-0.05	-0.06	0.16	-0.14	0.15	0.01	0.22	0.34	0.16	0.21	0.12	0.09
T_category	0.10	0.04	0.04	0.06	0.81	0.38	0.31	0.23	0.24	0.24	-0.01	-0.05	1.00	-0.13	-0.02	-0.14	-0.01	0.27	0.05	0.02	0.15	0.26	-0.06	0.35
ER	-0.03	0.08	-0.02	0.08	-0.14	-0.36	-0.40	-0.22	-0.36	-0.40	0.07	-0.06	-0.13	1.00	0.27	0.57	0.01	-0.45	-0.13	-0.05	-0.22	-0.28	0.00	0.00
ER_Allred_score	0.21	0.06	0.02	-0.02	0.08	-0.05	-0.07	0.02	-0.03	-0.19	0.18	0.16	-0.02	0.27	1.00	0.17	0.14	-0.28	-0.26	-0.09	0.05	0.01	0.11	0.06
PR	-0.18	0.03	-0.00	0.04	-0.10	-0.29	-0.30	-0.16	-0.27	-0.27	0.09	-0.14	-0.14	0.57	0.17	1.00	0.18	-0.27	-0.24	-0.13	-0.24	-0.19	-0.13	0.05
PR_Allred_score	0.04	0.03	0.04	0.03	0.06	-0.02	0.03	0.02	0.00	-0.00	0.15	0.15	-0.01	0.01	0.14	0.18	1.00	-0.00	-0.02	0.05	-0.12	-0.06	-0.08	-0.07
KI-67_LI_percent	-0.03	-0.10	0.04	0.04	0.24	0.52	0.55	0.30	0.51	0.56	0.06	0.01	0.27	-0.45	-0.28	-0.27	-0.00	1.00	0.09	0.06	0.08	0.35	0.08	0.18
HER2	0.01	-0.08	-0.02	-0.12	-0.00	0.15	0.13	0.04	0.12	0.16	-0.19	0.22	0.05	-0.13	-0.26	-0.24	-0.02	0.09	1.00	0.68	0.50	0.11	-0.08	-0.10
HER2_IHC	0.02	-0.02	-0.02	0.01	-0.02	0.17	0.12	0.03	0.15	0.11	0.07	0.34	0.02	-0.05	-0.09	-0.13	0.05	0.06	0.68	1.00	0.03	0.03	0.26	0.04
HER2_SISH	0.08	-0.11	0.07	0.03	0.11	0.24	0.21	0.22	0.23	0.07	0.05	0.16	0.15	-0.22	0.05	-0.24	-0.12	0.08	0.50	0.03	1.00	0.10	-0.17	0.11
HER2_SISH_ratio	0.01	-0.07	-0.09	-0.07	0.35	0.13	0.16	0.09	0.13	0.22	-0.12	0.21	0.26	-0.28	0.01	-0.19	-0.06	0.35	0.11	0.03	0.10	1.00	-0.16	-0.00
BRCA_mutation	0.11	-0.07	0.12	0.04	-0.08	0.05	-0.01	-0.22	0.04	0.04	0.15	0.12	-0.06	0.00	0.11	-0.13	-0.08	0.08	-0.08	0.26	-0.17	-0.16	1.00	0.09
N_category	0.07	-0.05	-0.03	0.27	0.29	0.37	0.25	0.27	0.29	0.05	0.32	0.09	0.35	0.00	0.06	0.05	-0.07	0.18	-0.10	0.04	0.11	-0.00	0.09	1.00

01 Preprocessing

결측치 처리

Tabular Data Preprocessing

1. **T_category**
clinical info에 설명된 정보를 근거하여 암의 장경이 0이고 DCIS_or_LCIS 여부가 1(DCIS)이면 0, 암이 장경 <= 20 이면 1, 암의 장경 <= 50이면 2, 암의 장경 > 50이면 3으로 대체. (4기는 암의 장경과 관계없이 흉벽이나 피부로 확장된 유방암 병변이 있는 경우인데 현재 데이터에서는 파악하기 어려우므로 암의 장경 기준으로만 0~3기 사이로 결측치 대체)

2. **암의 장경**
암의 장경과 T_category의 상관계수는 0.81이므로 매우 높은 수치. T_category별 암의 장경 중앙값을 구하여 대체. (train 데이터에서 계산한 통계량 사용)

3. **ER과 PR** : NG를 기준으로 NG가 1, 2면 ER과 PR을 1로, NG가 3이면 ER과 PR을 0으로 대체. NG를 기준으로 한 이유는 ER와 NG의 피어슨 상관계수가 -0.36, PR과 NG의 피어슨 상관계수가 -0.29로 음의 상관관계가 있다고 볼 수 있으며, 현재 데이터에서 ER 및 PR과 연관된 인자 중 NG가 비교적 결측치가 적기 때문에 이를 기준으로 ER과 PR을 대체하기 용이하다고 판단.

01 Preprocessing

결측치 처리

Tabular Data Preprocessing

4. ER_Allred_score

clinical info에 따르면 0~8까지의 범위를 갖는다. ER을 가진 세포가 얼마나 되는지의 비율 점수(PS)와 얼마나 강하게 ER을 나타내는지 강도 점수(IS)의 합산이다. ER_Allred_score가 0~2점이면 ER 음성이며, 3~8점은 ER 양성으로 분류된다. ER 양성 중에서도 3~4점은 약양성, 4~5점은 중양성, 7~8점은 강양성으로 나뉘게 된다.

이러한 정보를 근거로, ER이 음성인 데이터는 ER_Allred_score가 0~2점 사이, ER 양성인 데이터는 ER_Allred_score가 3~8점 사이라는 것을 역으로 추론할 수 있다. 이 때 현재 train 데이터 상에서 ER_Allred_score가 결측치가 아닌 데이터들은 3~8점까지 분포하고 있으므로, ER이 음성이면서 ER_Allred_score가 결측치인 데이터를 0~2점 중 세부적으로 구분할 필요없이 일괄 2로 대체하였다.

한편 ER이 양성이면서 ER_Allred_score가 결측치인 데이터는, ER 양성일 때의 ER_Allred_score의 중앙값인 7로 대체하였다. (train 데이터에서 계산한 통계량 사용)

c.f. reference1 : https://medicalcriteria.com/web/allred/

c.f. reference2 : https://www.researchgate.net/figure/Allred-score-total-score-TS-percentage-score-PS-intensity-score-IS-range-0_tbl1_257326052

01 Preprocessing

결측치 처리

Tabular Data Preprocessing

5. PR_Allred_score

ER_Allred_score과 마찬가지로 clinical info에 따르면 0~8까지의 범위를 가지며, PS와 IS의 합산이다.
이 때 현재 train 데이터 상에서 PR_Allred_score가 8을 초과하는 데이터(23, 54)가 존재하여 이를 이상치로 판단하고 일괄 8로 대체하였다.
한편 PR이 음성이면서 PR_Allred_score가 결측치인 데이터는 일괄 2점으로 대체하였다. 또한 PR이 양성이면서 PR_Allred_score가 결측치인 데이터는, PR 양성일 때의 PR_Allred_score의 중앙값인 6으로 대체하였다. (train 데이터에서 계산한 통계량 사용)
c.f. reference :ER_Allred_score reference와 동일

6. HER2

clinical info에 설명된 정보 및 추가적인 탐색을 통하여 얻은 정보에 따르면, HER2 세포성장인자 수용체의 경우 IHC 검사 결과 0, 1이면 음성, 2이면 중성, 3이면 양성에 해당하며, 이 IHC 검사에서 2 (중성)이 나오면 추가적으로 SISH 검사를 시행한다. SISH 검사 결과 0이면 음성, 1이면 양성으로 분류한다.
따라서 HER2_IHC와 HER2_SISH 컬럼을 바탕으로 HER2 결측치 값을 대체한다. 이후에도 남아있는 결측치는 일괄 0으로 대체한다.

01 Preprocessing

결측치 처리

Tabular Data Preprocessing

7. NG

우선 NG와 다른 인자와의 상관계수를 확인해보니, NG는 HG_score_2, HG, KI-67_LI_percent, HG_score_3, T_category 지표와의 상관계수가 높은 편이므로 이러한 인자들을 바탕으로 결측치를 대체하기로 한다.

1차적으로는 NG와 HG_score_2의 상관계수가 0.98로 매우 높기 때문에 NG의 결측치를 HG_score_2 값으로 대체하고, 2차적으로는 HG 값으로 대체한다. 한편, KI-67 지표의 경우 10%미만이면 low, 10~20%이면 medium, 20% 이상이면 high group으로 분류된다는 정보를 근거로, numerical value인 KI-67을 labeling index로 비닝 후 이 KI-67 labeling index별 NG의 최빈값을 구하여 결측치를 대체한다. 그리고 HG_score_3별 NG 최빈값으로 결측치를 대체하고, T_category별 NG 최빈값으로 결측치를 대체한다. (train 데이터에서 계산한 통계량 사용)

c.f. reference : https://link.springer.com/article/10.1007/s10549-022-06519-1

01 Preprocessing

결측치 처리

Tabular Data Preprocessing

8. **HG, HG_score_1~3**
HG 및 HG_score_2는 NG와의 상관계수가 높으므로 NG 값으로 대체한다. 이 때, NG의 범위는 1~3이고 HG 및 HG_score_1~3의 범위는 1~4이지만, clinical info에 따르면 HG 및 HG_score_1~3은 미세 침윤만 존재하는 경우 HG를 측정하지 않고 4로 분류하며 실제 점수는 1~3점 사이이므로 NG로 대체하였다. 한편 HG_score_1 및 HG_score_3의 결측치는 HG 값으로 대체한다.

9. **KI-67_LI_percent**
현재 데이터에서 KI-67은 NG와의 상관계수가 0.52이며 논문을 찾아보니 KI-67은 NG, HER2와 상관관계가 높으므로 NG별 중앙값으로 대체한다. (train 데이터에서 계산한 통계량 사용)
c.f. reference : https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4854054/

10. **BRCA_mutation**
clinical info에 따르면 검사를 시행하지 않은 경우가 결측치므로, 음성 및 양성 결과와 구분하기 위해 결측치를 일괄 -1로 대체한다.

11. **기타** : 일괄 0으로 대체.

01 Preprocessing

Feature Generation

Tabular Data Preprocessing

1. **수술연도 Feature Generation**
수술연월일 데이터에서 수술연도를 추출하여 추가하였다.

2. **Subtype Feature Generation**
Subtype은 유방암 예후에 중요한 인자가 된다. 호르몬 양성 + HER2 양성은 예후가 좋은 편이며, 호르몬과 HER2가 모두 음성인 삼중음성은 예후가 좋지 않다. 따라서 이러한 정보를 모델이 학습할 수 있도록 Subtype 컬럼을 새롭게 추가하였다.

3. **최종적으로 학습에 사용한 Features**
['나이', '진단명', '암의 위치', '암의 개수', '암의 장경', 'NG', 'HG', 'HG_score_1', 'HG_score_2', 'HG_score_3', 'DCIS_or_LCIS_여부', 'DCIS_or_LCIS_type', 'T_category', 'ER', 'ER_Allred_score', 'PR', 'PR_Allred_score', 'KI-67_LI_percent', 'HER2', 'BRCA_mutation', '수술연도', 'Subtype']

02 Modeling

Multiple Instance Learning

01 Multiple Instance Learning

02 시도한 것들

02 Modeling

Multiple Instance Learning

Multiple Instance Learning

1. Multiple Instance Learning 의 필요성

병리 슬라이드는 보통 사이즈가 큰데 일반적인 computer vision 모델에 사용하기 위해 해상도를 줄이게 될 경우 제대로 된 학습이 어려움. 또한 전체 병리 슬라이드 이미지를 학습하기에 자원이 많이 소요됨.

따라서 Multiple Instance Learning 을 사용.

2. Multiple Instance Learning 의 구조

1) 전체 병리 슬라이드를 tiles로 나눔. 이 때 개별 tile은 모델에서 instance가 되며, tiles의 집합이 bag of instances라고 할 수 있음.
2) Feature Extractor에서 개별 tile의 feature를 추출.
3) Aggregator에서 tiles의 features를 모아 bag-level feature로 합침.
4) bag-level에서 prediction을 진행.

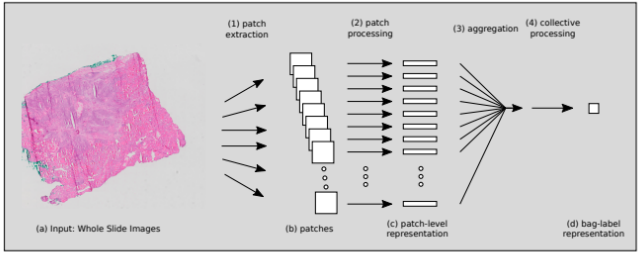


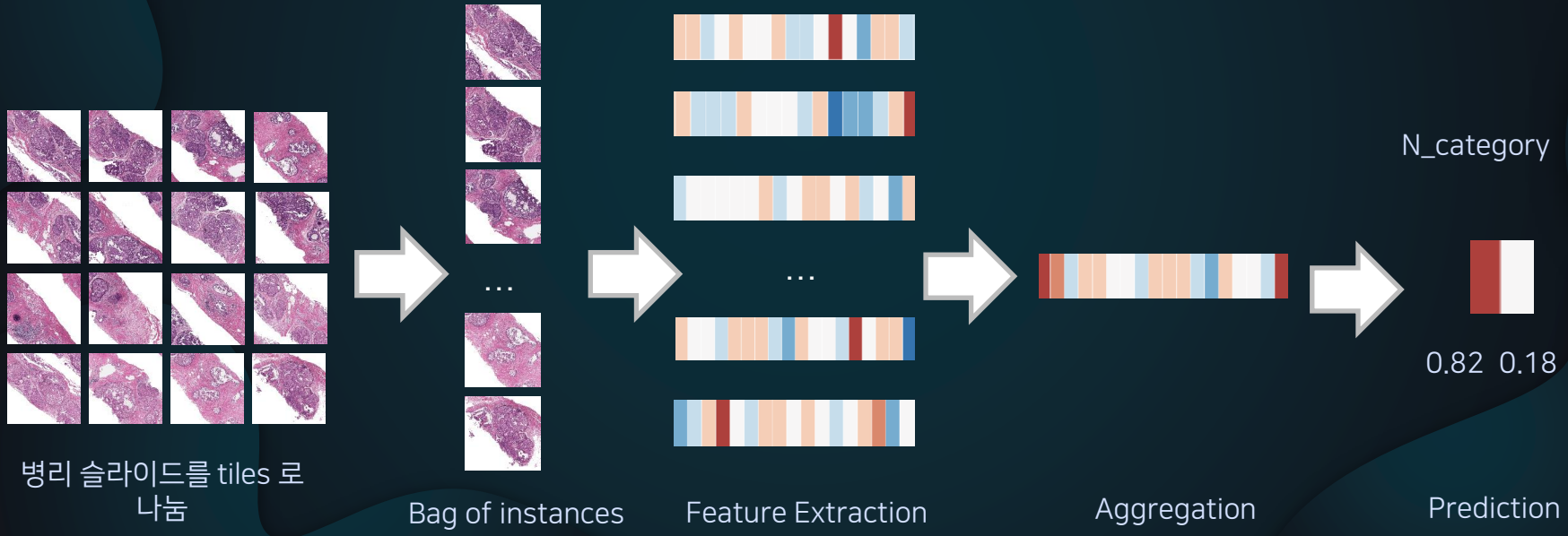
Figure 2: High-level perspective on MIL applied to WSIs. From the input images (a), patches (b) are extracted, followed by patch-level processing resulting in patch-level representations (c) and aggregation (several patch-level features to a single bag-level feature) and collective processing resulting in bag-level representations (d).

c.f. Multiple Instance Learning for Digital Pathology: https://arxiv.org/pdf/2206.04425.pdf

02 Modeling

Multiple Instance Learning

3. Multiple Instance Learning 현재 대회에서의 적용



02 Modeling

시도한 것들

Multiple Instance Learning

1. Backbone : swin_tiny_patch4_window7_224 사용
그외 efficientnet_b0, efficientnet_b3, densenetblur121d, deit3_base_patch16_384, swin_large_patch4_window12_384를 사용해봤는데 swin_tiny_patch4_window7_224 성능이 가장 좋았음. 일단 기존 병리 슬라이드를 tiles로 분할하여 확대된 조직 이미지를 사용하기 때문에 이미지 사이즈가 224x224 더라도 세부적인 학습이 가능한 것으로 판단됨. 또한 CNN 모델보다 vision transformer 모델이 조직 이미지 상의 특징을 깊이있게 탐색했을 것으로 판단됨. 한편 swin_large_patch4_window12_384로 384x384 이미지 사이즈로 실험해보려고 했으나 모델이 무거워서 n_instances 16을 학습하지 못하여 n_instances 8로 실험해봤는데 효과적이지 않았음. n_instances 개수가 줄어서 학습에 필요한 부분이 누락되었을 것으로 생각됨.

2. n_instances 16
n_instances 8, 16, 25 으로 실험해보았고 16이 가장 성능이 좋았음. 8개 tiles로는 모델이 분류를 위해 필요한 부분들을 온전히 다 파악하지 못하는 것 같고, 25개 tiles는 불필요한 부분까지 학습했거나 instances 개수가 많아 연산량이 늘어서 모델이 제대로 학습하지 못한 것으로 판단됨.

02 Modeling

시도한 것들

Multiple Instance Learning

3. 각 instance에서 feature extraction을 통해 도출하는 embedding size 512로 변경
swin_tiny_patch4_window7_224 에서 기존 feature dimension 768로도 실험해보았고, 256으로 줄이기도 해봤는데 512일 때 성능이 미세하게 더 좋았음.

4. image transforms 시 custom normalization
기존 imagenet mean과 std 기반으로 normalize하는 대신 현재 병리 슬라이드 tiles의 pixels의 mean과 std를 별도로 계산하여 해당 값으로 custom normalization한 것이 더 성능이 좋았음.
mean = [0.8959, 0.8123, 0.8792], std = [0.1211, 0.2004, 0.1296]

5. Tissue segmentation 후 crop한 이미지로 tile generation 후 학습
원본 이미지로 tiles을 만들어서 학습할 경우, noise가 많아 MIL Model에서 학습이 제대로 되지 않았음.
Tissue segmentation 후 crop하여 tiles을 생성하여 학습하니 성능이 큰 폭으로 상승하였음.

02 Modeling

Multi-model (Image, Tabular) Learning

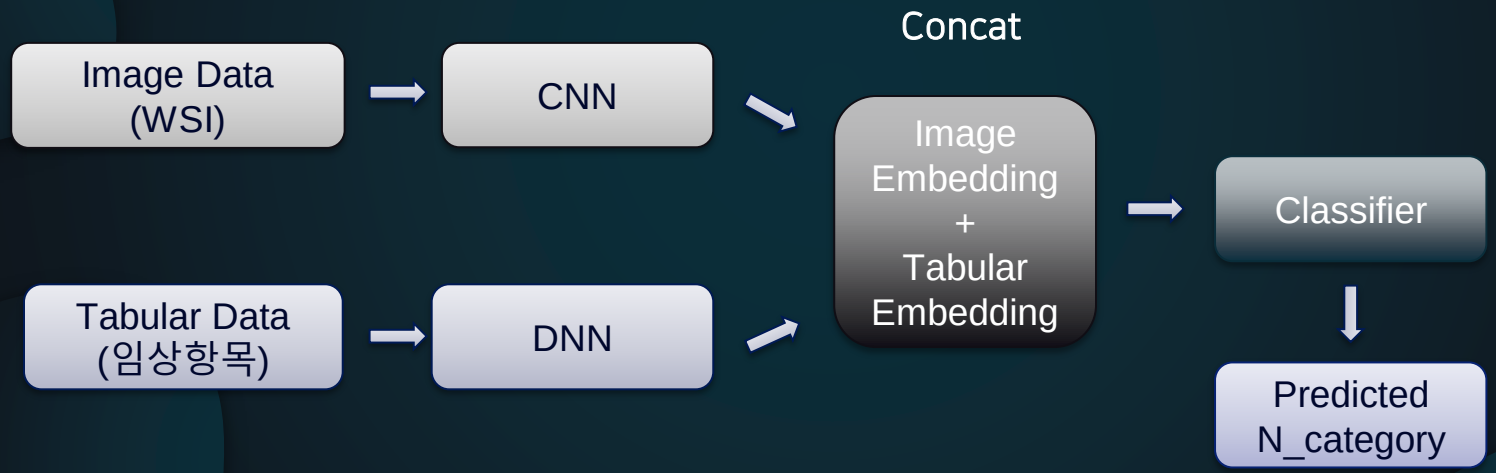
01 Multi-modal Learning

02 시도한 것들

02 Modeling

Multi-modal Learning

Multi-modal Learning 현재 대회에서의 적용



02 Modeling

Multi-modal Learning

시도한 것들

1. **Backbone : densenetblur121d 사용**
그 외 efficientnet_b0과 efficientnet_b3_ns 를 사용해봤는데 densenetblur121d 성능이 가장 좋았음. 더 다양한 모델들로 탐색해봤으면 좋았을 것이라는 아쉬움이 있음.

2. **전체 병리 슬라이드 이미지를 1024 x 1024 사이즈로 학습**
이 대회의 원본 병리 슬라이드 해상도가 비교적 좋진 않았기 때문에 tiles를 기반으로 한 MIL model 뿐만 아니라 전체 병리 슬라이드 이미지를 사용한 모델로 학습할 필요도 있다고 판단. 그 대신 이미지 사이즈를 많이 줄이기보다는 전체 병리 슬라이드 이미지를 1024x1024 사이즈로 학습. 256 x 256이나 512 x 512로도 학습해봤으나 1024 x 1024가 더 효과적이었음.

3. **Image Embedding size와 Tabular Embedding size를 각각 512으로 설정**
이미지 모델에서 기존 feature dimension인 1024로도 실험해보았고, 256으로 줄여보기도 했는데 512가 가장 성능이 좋았음. 정형 데이터 모델에서는 64, 256, 512로 embedding size를 변경해봤는데 512가 가장 성능이 좋았음.

02 Modeling

Tree Based Boosting Algorithm

01 Tree Based Boosting Algorithm

02 시도한 것들

02 Modeling

Tree Based Boosting Algorithm

1. **모델 선택 : XGBoost 선택**
Boosting model은 학습이 빠르고 정형 데이터를 파악하는 것에 있어서 DNN보다 Tree Based Boosting Algorithm이 현재까진 더 통용되고 있고 효과적. Tree Based Boosting Algorithm 중에서도 XGBoost뿐만 아니라 LGBM과 Catboost도 시도해보았는데 LGBM은 train 데이터에 과적합되는 경향이 있었고 Catboost는 XGB나 LGBM에 비해 성능이 좋지 않아 최종적으로 XGBoost 모델로 정형 데이터를 학습.

2. **하이퍼 파라미터 최적화 : Optuna**
Optuna를 통하여 하이퍼 파라미터 최적화를 진행. {'subsample': 1.0, 'colsample_bytree': 0.6, 'gamma': 0.2103436350297136, 'reg_lambda': 0.212428798782124 67, 'reg_alpha': 6.038789883743567, 'max_depth': 13, 'min_child_weight': 3} 으로 학습.

03 Cross Validation

Stratified 5-Fold Validation

01 Stratified K Fold

02 split 기준 선택

03 Cross Validation

Stratified 5-Fold Validation

1. Cross Validation 방식 : StratifiedKFold

현재 대회에서는 N_category (림프 전이) 여부를 0과 1로 분류해야 함. 따라서 target class가 train 데이터와 valid 데이터에 골고루 존재할 수 있도록 Stratified 방식으로 폴드셋을 split함.

2. Stratified 기준

이 때 target인 N_category 만을 기준으로 층화 추출을 진행하면 폴드셋별로 성능 편차가 심하다는 문제가 있었음. 그래서 데이터의 특징을 골고루 학습할 수 있도록 N_category 뿐만 아니라 다른 인자를 합쳐서 2개의 기준으로 Stratify 진행. 이 때 기준 선택의 조건은 데이터 폴드셋별 점수 편차가 크지 않은 것, 최종적인 평균 성능이 좋은 것으로 탐색.
그 결과 N_category 단독 기준 < N_category + Subtype (호르몬 수용체, HER2 결합 컬럼) 기준 < N_category + T_category 기준 < N_category + NG 기준 < N_category + HG 기준으로 더 성능이 좋아서 최종적으로 **N_category + HG** 기준으로 층화 추출 진행.
참고로 정형 데이터만 학습할 때에는 N_category 에 Subtype이나 T_category를 합친 기준으로 나눈 폴드셋들의 점수 편차도 크지 않고 성능도 좋았으나, 이미지를 학습시킬 때에는 N_category + HG 기준으로 나눈 폴드셋이 전반적으로 점수 편차가 크지 않고 최종적인 평균 점수도 가장 좋았음. NG나 HG가 병리 슬라이드 이미지와 직접적으로 연관된 인자이기 때문에 이를 기준으로 데이터를 나누면 폴드셋별로 이미지 데이터 특징이 비교적 균일하게 분포하게 되는 것으로 생각됨.

04 Ensemble

Stacking

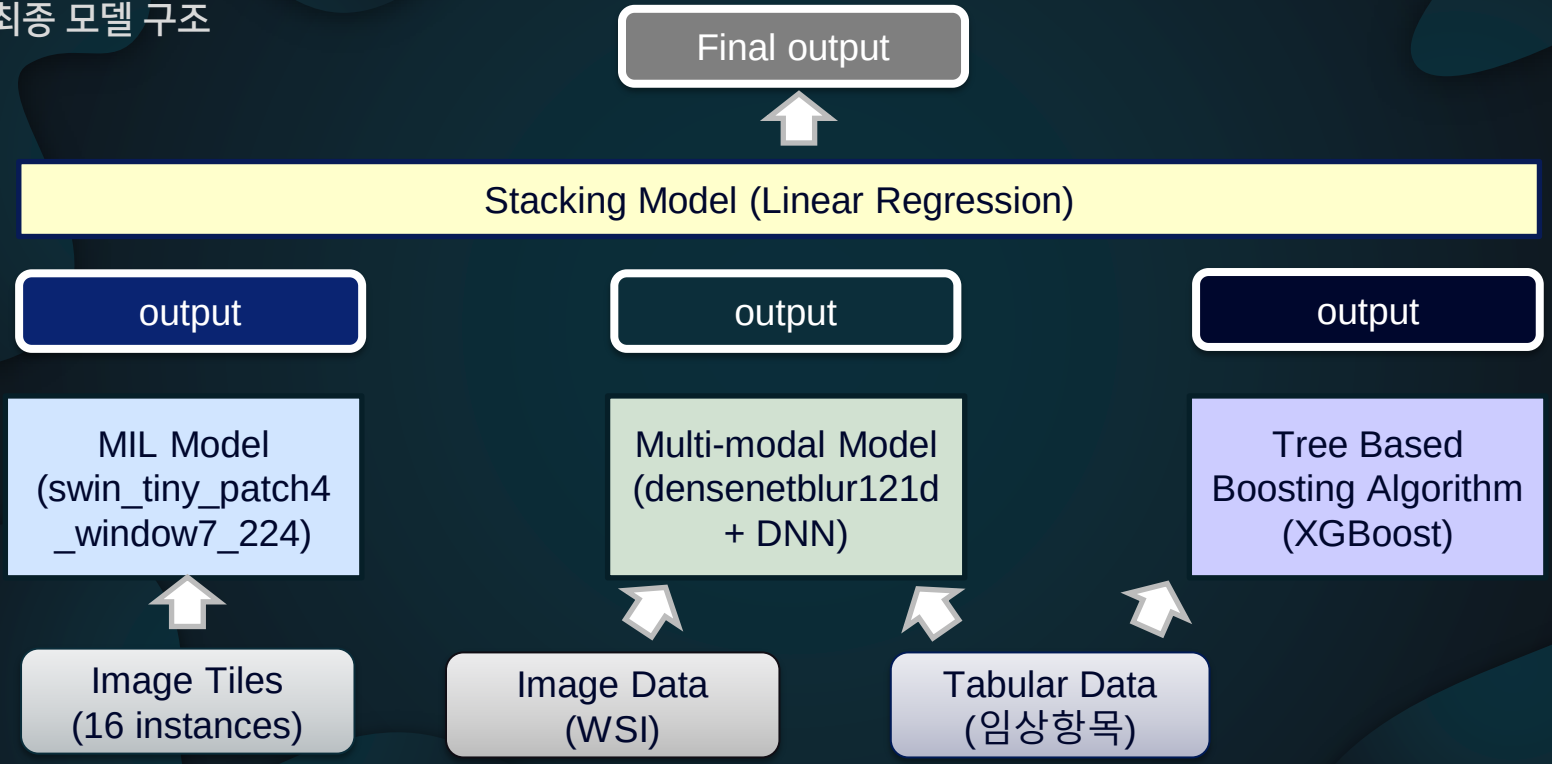
01 최종 모델 구조

02 Stacking

04 Ensemble

최종 모델 구조

최종 모델 구조



04 Ensemble

Stacking

1. Ensemble
최종적으로 MIL Model, Multi-modal (Image, Tabular) Model, Xgboost Model 3가지 모델에서 도출한 결과를 앙상블.

2. Stacking Ensemble
블렌딩과 스태킹, 보팅을 사용해본 결과 가장 성능이 좋은 스태킹 기법을 사용. Meta classifier로 가벼운 linear regression 모델을 선택.

05 Wrap up

패키지 실행 방법

1. 라이브러리 임포트	pip install -r requirements.txt
2. Preprocessing	python main.py config/preprocessing_config.yaml preprocessing
3. MIL 모델 training	python main.py config/mil_config.yaml train
4. MIL 모델 inference	python main.py config/mil_config.yaml inference
5. Multi-modal 모델 training	python main.py config/convnet_tabular_config.yaml train
6. Multi-modal 모델 inference	python main.py config/convnet_tabular_config.yaml inference
7. XGB 모델 training	python main.py config/xgb_config.yaml train
8. XGB 모델 inference	python main.py config/xgb_config.yaml inference
9. Stacking Ensemble	python main.py config/ensemble_config.yaml ensemble

05 Wrap up

Environment

OS : Ubuntu 18.04.6 LTS

Python : 3.8.16

GPU : A100-SXM4-40GB

패키지 파일 구조

```
├── ./config
│   ├── ./config/convnet_tabular_config.yaml
│   ├── ./config/ensemble_config.yaml
│   ├── ./config/mil_config.yaml
│   ├── ./config/preprocessing_config.yaml
│   └── ./config/xgb_config.yaml
├── ./data
│   ├── ./data/clinical_info.xlsx
│   ├── ./data/sample_submission.csv
│   ├── ./data/test.csv
│   ├── ./data/test_imgs
│   ├── ./data/test_imgs_crop
│   ├── ./data/test_preprocessed.csv
│   ├── ./data/test_tiles
│   ├── ./data/train.csv
│   ├── ./data/train_imgs
│   ├── ./data/train_imgs_crop
│   ├── ./data/train_preprocessed.csv
│   └── ./data/train_tiles
├── ./ensemble.py
├── ./image_preprocessing.py
├── ./log
├── ./main.py
├── ./metrics.py
├── ./model
│   ├── ./model/densenetblur121d_image1_1024_tabular
│   ├── ./model/swintinypatch4window7_mil16_224
│   └── ./model/xgboost_tabular
├── ./README.md
├── ./requirements.txt
├── ./settings.py
├── ./submission
├── ./tabular_preprocessing.py
├── ./torch_dataset.py
├── ./torch_model.py
├── ./torch_trainer.py
├── ./transforms.py
├── ./utils.py
└── ./xgb_trainer.py
```

감사합니다.

임효정 lim.gadi@gmail.com