Multimodal Early Fusion Transformer for stock price prediction

Tae-Won Lee

leetee95@gmail.com

19th Oct, 2022

To Do List from Previous Seminar

이번 주

- 졸업논문 Feedback Revision
- 신경망의 세부적인 그림

Feedback Revision

- Comment

Several recent financial studies combined various data with stock price data [13]. The model combined with CNN and LSTM was proposed in [31] to extract the feature of limit order book and stock price data. In this work, the CNN module of the model extracts the feature in the limit order book, and the LSTM module of the model extracts the feature in stock price data. The information fusion-based genetic algorithm (GA) was proposed to predict stock prices and trends using LSTM in [32]. The proposed GA and LSTM aim to optimize the parameters of an LSTM, select a set of features, and is executed with inter-intra crossover and adaptive mutation. The heterogeneous information fusion method was proposed in [33] for information fusion on the Web and financial news for stock market prediction. The authors in [34] addressed the problem of technical analysis information fusion in improving stock market index-level prediction. The multimodal deep learning architecture was proposed in [33], where the architecture learns the cross-correlation of the United States and Korean stock prices. These studies commonly reported that the combination of various data sources can contribute to the improvement of prediction accuracy.

Increase the level of detail for each item. Do not use the same style of phrases. "is proposed. ... is proposed. ... is proposed. Do use 'was' instead of 'is' from this section.

- Revision

Several recent financial studies combined various data with stock price data [13]. An event-driven approach was used to predict stock price prediction with stock price data in [32]. The event-driven method was elaborated by extracting the stock-related event from the news titles. Moreover, the method enhancing the joint impacts was also devised by calculating the two stocks' similarity using their p-change values and Pearson correlation coefficient. The information fusion can be enhanced by conducting parallel operations that contain inter-intra crossover and adaptive mutation in the genetic algorithm (GA) [33]. The proposed information fusion-based GA approach can optimize the parameters of an extended short-term memory stock price prediction model and selects a set of features. Historical stock quantitative data, social media data, and web news data could be used by expressing the information as a matrix and tensor to predict stock market prediction [34]. The stock quantitative feature matrix and stock correlation matrix were made using the stock's quantitative and social media data. The stock movement tensor was built by events and sentiments extraction from news articles and social media stock quantitative feature matrix and the stock correlation matrix are factorized, and the stock movement tensor was decomposed to predict stock price prediction. The authors in the work of [35] addressed the problem of technical analysis information fusion in improving stock market index-level prediction. Multiple predictive systems based on different categories of technical analysis metrics are devised. The system has multiple prediction models with different technical analysis metrics data categories. Each prediction model was based on an ensemble of neural networks with particle swarm intelligence for parameter optimization The authors in the work of [36] proposed the multimodal deep learning architecture where the architecture learns the cross-correlation of the United States and Korean stock prices. These studies commonly reported that combining various data sources could improve

Feedback Revision

- Comment

In this study, we consider the problem of the stock direction prediction [?, ?, ?] that predicts whether the stock prices will rise or fall in the future because *Why do you focus on this problem? Is it important?*. Specifically, the proposed method predicts the next day's stock up and down direction using the

B. MULTIMODAL EARLY FUSION TRANSFORMER

An accurate stock price prediction *From now on, do not use "stock price prediction." Do use "stock direction prediction."* is a challenging problem because of the randomness in the

- Revision

III. MATERIALS AND METHOD

In this study, we consider the problem of the stock direction prediction [37] that predicts whether the stock prices will rise or fall in the future because it gives a guide to investors to buy a stock before the price rises or sell it before its value declines [38]. Specifically, the proposed method predicts the

B. MULTIMODAL EARLY FUSION TRANSFORMER

An accurate stock direction prediction is a challenging problem because of the randomness in the stock price data [2]. To solve this problem, we propose a multimodal early fusion neural network that fuses various information sources internally. Considering the external factors in stock direction prediction is crucial because the stock price is affected by many external factors [40]. Late fusion and early fusion are

Feedback Revision

- Comment

Give the notation used subsequently in this paper. Add the formulation about the input data source. What is the size of each modality? Do not make your own notation. Do use the notations from other papers. We select feature Concatenation method in early fusion because it is one of the efficient and effective method to fuse multiple information sources.

$$M = Concat(m_1, m_2, m_3), \quad (1)$$

where M is modality matrix concatenated by time, m_1 is the stock price modality, m_2 is the months and day of the week modality, and m_3 is the macroeconomic indicator modality.

It can be different which day affects to the next day in predicting the stock price. We use the positional encoding module to let the model know the position information of the data. *(What is the meaning of two equations? what is the difference?*

$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{model}}) \quad (2)$$

and

$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{model}}) \quad (3)$$

where pos is the position and i is the dimension. d_{model} is the feature dimension of modality data concatenated by time.

- Revision

We select feature Concatenation method in early fusion because it is one of the efficient and effective method to fuse multiple information sources.

$$M = Concat(x_1, x_2, x_3), \quad (1)$$

where M is modality matrix concatenated by time, $x_1 \in \mathbb{R}^{l \times F_1}$ is the stock price modality, $x_2 \in \mathbb{R}^{l \times F_2}$ is the months and day of the week modality, and $x_3 \in \mathbb{R}^{l \times F_3}$ is the macroeconomic indicator modality. l is sequence length. F_1, F_2 and F_3 are feature sizes of the stock price modality, the months and day of the week modality, the macroeconomic indicator modality, respectively.

2) Positional Encoding

It can be different which day affects the next day's stock price prediction. We use the positional encoding module to let the model know the position information of the data. The sine and cosine functions are used to express different frequencies [26].

Feedback Revision

- Comment

5) Compression Feed-Forward Networks

First, feature compress Feed-Forward Networks extract the information between features extracted by Multi-Head Attention. Second, sequence compress Feed-Forward Networks extract the information between the time. *(What is the difference between two equations?)*

$$FFN_f(x) = max(0; xW_f + b_f) \quad (7)$$

and

$$FFN_s(x) = max(0; xW_s + b_s) \quad (8)$$

where W_f and b_f are the weight and bias of feature compress Feed-Forward Networks. W_s and b_s are the weight and bias of sequence compress Feed-Forward Networks.

IV. EXPERIMENTAL RESULTS

A. EXPERIMENTAL SETTINGS

(Move this part to the "data preprocessing part" and eliminate duplicated explanations.) We conducted our experiment by setting three modalities: stock price modality, months and day of the week modality, and macroeconomic indicator modality. In stock price modality, we used the KDD17 public dataset [31] to add to the reliability of our experiment. The features of the KDD17 dataset are open, close, high, low, volume, and adj close. And we also added the ten technical indicators [39] and two additional features in the stock price modality through feature engineering to extract significant features in stock price data. Table1 shows the details of the stock price modality. In the months and day of the week modality, the months and day of the week features are extracted using One-Hot Encoding because months is a categorical variable with 12 classes, and the day of the week is also a categorical variable with seven classes. The extracted one-hot encoding vector can be expressed as $[0, 1, \cdots, 0]$. In the macroeconomic indicator modality, we selected six features such as NASDAQ100, US 2-year, 10-year, 30-year Bond Yield, US Dollar Index, and WTI Oil Price. We used only the close column in the data even if the data have plural columns such as open, high, low, close, and volume.

(Add the characteristics of KDD17 dataset here.)

- Revision

5) Compression Feed-Forward Networks

First, feature compression feed-forward networks extract the information between features extracted by Multi-Head Attention. After getting through the feature compression feed-forward networks, the feature dimension can be one. Second, time sequence compression feed-forward networks extract the information between the time and the time sequence dimension can be one.

A. DATA PREPROCESSING

We conducted our experiment by setting three modalities: stock price modality, months and day of the week modality, and macroeconomic indicator modality. In stock price modality, we used the KDD17 public dataset [31] to add to the reliability of our experiment. The features of the KDD17 dataset are open, close, high, low, volume, and adj close. And we also added the ten technical indicators [39] and two additional features in the stock price modality through feature engineering to extract significant features in stock price data. Table1 shows the details of the stock price modality. In the months and day of the week modality, the months and day of the week features are extracted using One-Hot Encoding because months is a categorical variable with 12 classes, and the day of the week is also a categorical variable with seven classes. The extracted one-hot encoding vector can be expressed as $[0, 1, \cdots, 0]$. In the macroeconomic indicator modality, we selected six features such as NASDAQ100, US 2-year, 10-year, 30-year Bond Yield, US Dollar Index, and WTI Oil Price. We used only the close column in the data even if the data have plural columns such as open, high, low, close, and volume.

IV. EXPERIMENTAL RESULTS

A. EXPERIMENTAL SETTINGS

We used KDD17 datasets [39] and macroeconomic indicators on stock direction prediction. The macroeconomic indicators were collected on Investing.com. The length of historical price in KDD17 datasets and macroeconomic indicators is the 2518days ranging from Jan-01-2007 to Jan-01-2016. The KDD17 dataset has 50 stocks in U.S. markets. This paper's macroeconomic indicators include NASDAQ100, U.S. 2-year, 10-year, and 30-year Bond Yield, U.S. Dollar Index, and WTI Oil Price.

Feedback Revision

- Comment

review on three comparison models are as follows:

- CNN-Attention BiLSTM [1]: *(Wenjie Lu et al.) <- I have warned you.* proposed a model that combines CNN and Attention Bi-LSTM to predict stock price prediction. CNN is used to extract local perception and improve the efficiency of model training by reducing the number of parameters. Attention Bi-LSTM is used to learn sequential features of the stock data. They used the Shanghai Composite Index (000,001) stock as a dataset.
- Adversarial LSTM [30]: *(Fuli Feng et al.)* proposed an adversarial training method for stock data using Attention LSTM. The adversarial training method is a module that captures the stochastic property in stock data. The adversarial training method has been used for computer vision tasks at the data level. Still, this paper concatenates feature-level perturbations during training to express inherent stochastic properties in stock data.
- DTML [29]: *(Jaemin Yoo et al.)* proposed a DTML model that is combined with the Attention LSTM and Transformer model. They used the Attention LSTM model to extract the sequential feature in the stock data and concatenated the model output. And Transformer model is used to learn the correlation between the plural stocks in the concatenated Attention LSTM feature.

- Revision

- CNN-Attention BiLSTM [1]: CNN-Attention BiLSTM model combines CNN and Attention Bi-LSTM to predict stock price prediction. CNN is used to extract local perception and improve the efficiency of model training by reducing the number of parameters. Attention Bi-LSTM is used to learn sequential features of the stock data. They used the Shanghai Composite Index (000,001) stock as a dataset.
- Adversarial LSTM [31]: The adversarial training method is a module that captures the stochastic property in stock data. The adversarial training method has been used for computer vision tasks at the data level. Still, this paper concatenates feature-level perturbations during training to express inherent stochastic properties in stock data to overcome the stochastic property in stock price data for stock price prediction.
- DTML [30]: DTML model is combined with the Attention LSTM and Transformer model. They used the Attention LSTM model to extract the sequential feature in the stock data and concatenated the model output. And Transformer model is used to learn the correlation between the plural stocks in the concatenated Attention LSTM feature.

where TP, TN, FP, and FN refer to True Positives, True Negates, False Positives, and False Negates. *(Move this part to the proposed method section.)* The output of the final node in neural network get through the sigmoid function. The sigmoid function is the non-linear function that makes the input value in the range of $[0, 1]$. The output value of

5

6) Model output

The output of the final node in neural network get through the sigmoid function. The sigmoid function is the non-linear function that makes the input value in the range of $[0, 1]$. The output value of the sigmoid function is used for the input of Binary Cross Entropy Loss with the target value. Model parameters are updated using the value of the Binary Cross Entropy Loss.

Feedback Revision

- Comment

Table 5 shows experimental results for the entire data in terms of accuracy. *(Add the explanation how to read the Table 5.)* The Table 6 shows the abbreviation of each sector. The proposed method, 27 out of 50 stock data, showed the best accuracy compared to comparison models. Compared to the comparison model, the stock that the proposed method

- Revision

Table 5 shows experimental results for the entire data in terms of accuracy. The table consists of a ticker, sector, proposed method's experiment results, and three comparison model's experiment results. The ticker represents the name of a certain stock. The sector in stock is a unit consisting of similar industries. The experiment results are expressed as an average accuracy and standard deviation. The Table 6

FIGURE 1: The overview of the proposed early fusion method. *(Increase the detailedness about the internal information processing.)*

FIGURE 1: The overview of the proposed early fusion method. The proposed method comprises the feature concatenation, transformer encoder, and classifier layer.

$$BCE(x) = -\frac{1}{N}\sum_{i=1}^{N} y_i log(h(x_i)) + (1-y_i)log(1-h(x_i))$$

where N, y_i, and $h(x_i)$ are the number of data, 0/1 binary target value is the prediction of the proposed model, respectively. *(Incomplete explanation. Where is the definition of h function here?)*

$$BCE(x) = -\frac{1}{N}\sum_{i=1}^{N} y_i log(h(x_i)) + (1-y_i)log(1-h(x_i)) \quad (9)$$

where N, y_i, and $h(x_i)$ are the number of data, 0/1 binary target value is the prediction of the proposed model, and the final output of the model gets through the sigmoid function, respectively.

Feedback Revision

- Comment

product is divided by $\sqrt{d_k}$. In this paper, *(Weird explanation.) which parts of the data are more interrelated may be important because learning the inter-correlation between data is one of the objectives.*

Figure 3 shows the result of the Bonferroni-Dunn test. (Add explanation how to read Figure 3 and the proposed method outperforms the counterparts here.)

methods. *Summarize the superiority of the proposed method based on the experimental results here.*

- Revision

product is divided by $\sqrt{d_k}$. In this paper, it is important to know what parts of the data are related to each other because learning the inter-correlation between data is one of the objectives.

Bonferroni-Dunn test. In the figure, the three comparison models are farther to the left than the CD (=0.61812) compared to the proposed model. The proposed method achieved the best average rank and outperformed all the comparison methods.

outperformed all the comparison methods. The quantitative analysis confirmed that the proposed method has a numerical difference compared to the comparison model. As a result of the statistical tests, it was found that the proposed model had a statistically significant difference compared to the comparison models. Therefore, the proposed method outperformed the comparison models regarding classification accuracy.

Feedback Revision

- Comment

B. COMPARISON RESULTS

(Put the table showing the result of shapiro test here.)

Table 5 shows experimental results for the entire data in terms of accuracy. The table consists of a ticker, sector,

methods to combine plural data in Multimodal deep learning. *(Weird explanation.) Whereas the late fusion method extracts intra-modality correlation more efficiently than the early fusion method, the early fusion method extracts inter-modality correlation more efficiently than the late fusion method [36]. (Weird explanation.) In this paper, our purpose is to develop a model that learns the relation between modalities well because data coming from other sources can affect each other in the financial domain.* Therefore, the early fusion method is selected in this paper. The transformer is a Natural Language

- Revision

TABLE 5: Summary of the Shaprio-Wilk statistic $W(N = 50)$ and Critical Value in terms of Accuracy measure

model	W	Critical Value ($\alpha = 0.05$)
Proposed method	0.9485	0.947
CNN-Attention BiLSTM [1]	0.9791	
Adversarial LSTM [31]	0.9725	
DTML [30]	0.9727	

Feedback Revision

- Comment

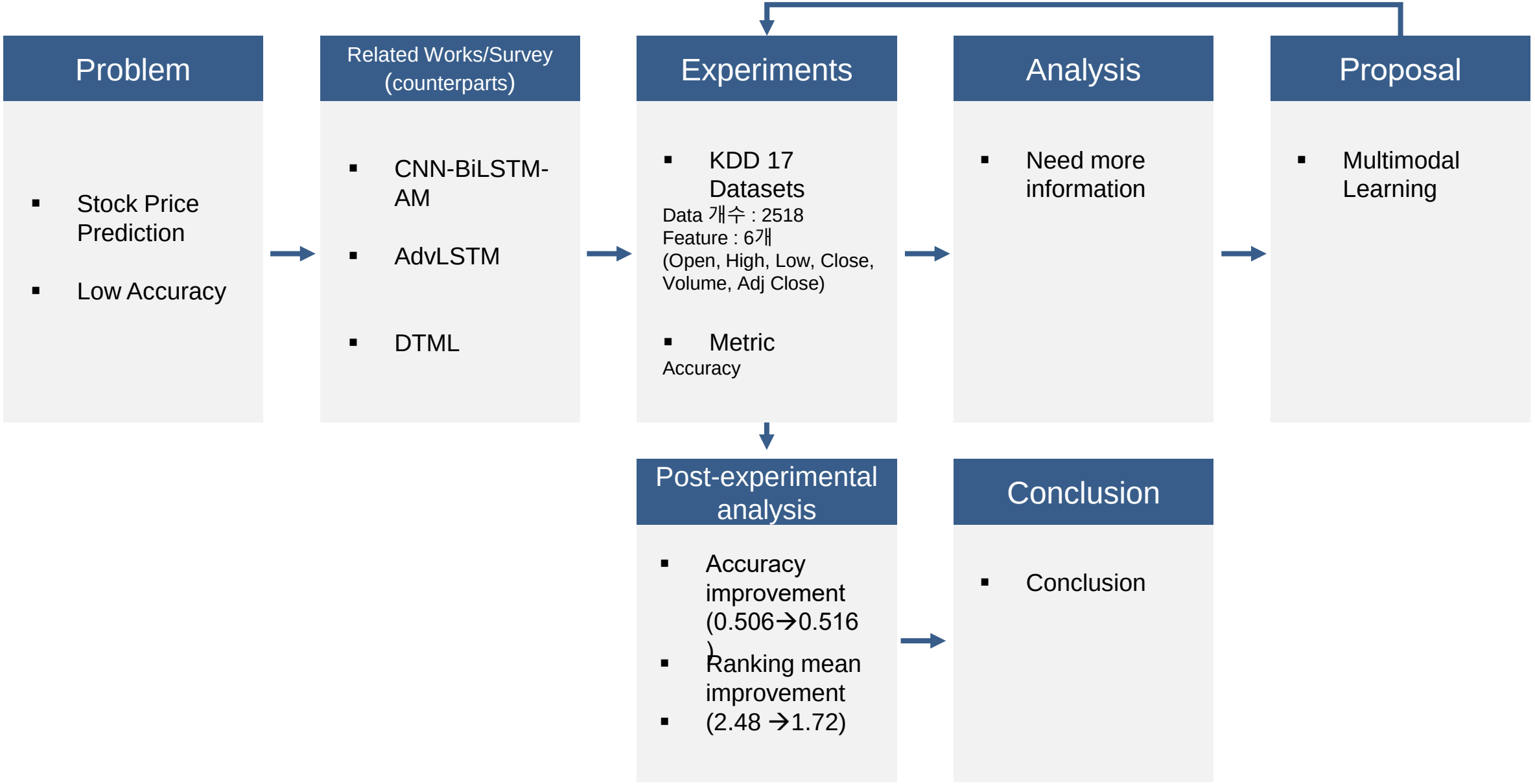
map. *(This explanation should be enhanced.)* In (a) early fusion, we can see that each modality learns by referring to each other. However, in (b) late fusion, each modality is separated and passes through the scaled dot-product attention. Consequently, the early fusion method learns inter-modality correlation better than the late fusion method because all the modalities get through the scaled dot-product attention together, which learns the relationship between data in the early fusion method.

V. CONCLUSION

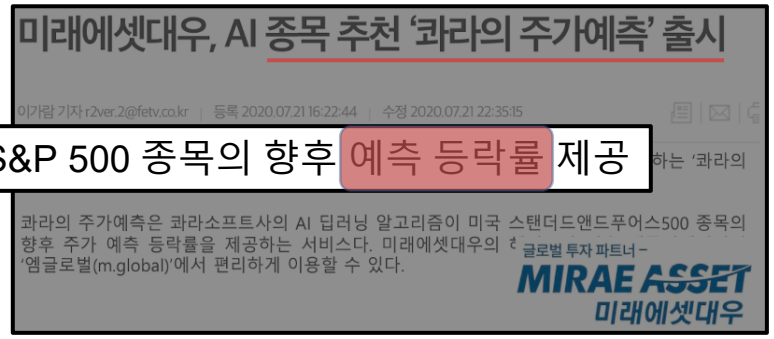
(Is it necessary?) Stock direction prediction is tremendous influence and utility in the financial domain. Despite many researchers' efforts to solve the problem, they still have experienced a problem because of the difficulty of the stock direction prediction. An enormous ripple effect can come from financial data mining if it is possible to predict accurate stock prices. We tried to solve this problem by fusing various data. In this study, we proposed multimodal early fusion methodology *(what is the meaning of this?)* to carry out the relevant task effectively. Our model outperformed compared to the comparison model and achieved statistically significant results. Specifically, BLAH *(Add the detailedness on this.) (Is it necessary?)* Hereafter, In addition to the data used in this paper, we could expect better research results if additional data exploratory research is applied in financial data mining. Moreover, data fusion methodology considering the trait between different data is also important in multimodal stock direction prediction. *Discuss the future research direction here.*

- Revision

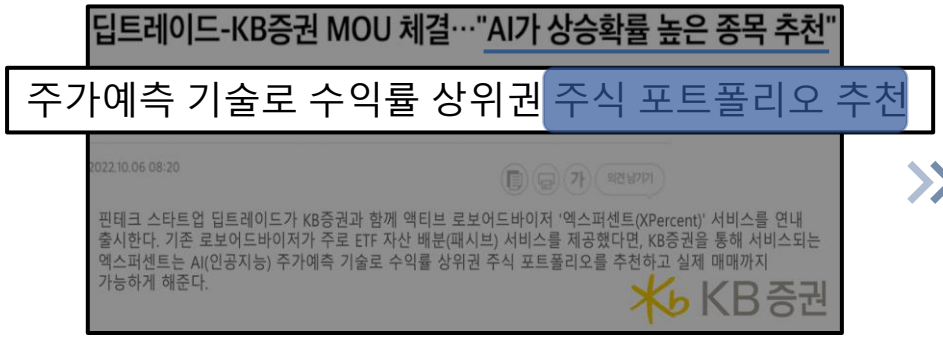
Overall Progress



Stock Price Prediction



Stock prospect suggestion



Portfolio recommendation

Core Task

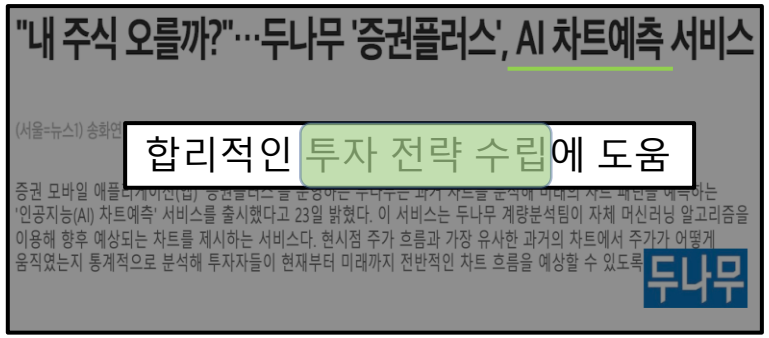
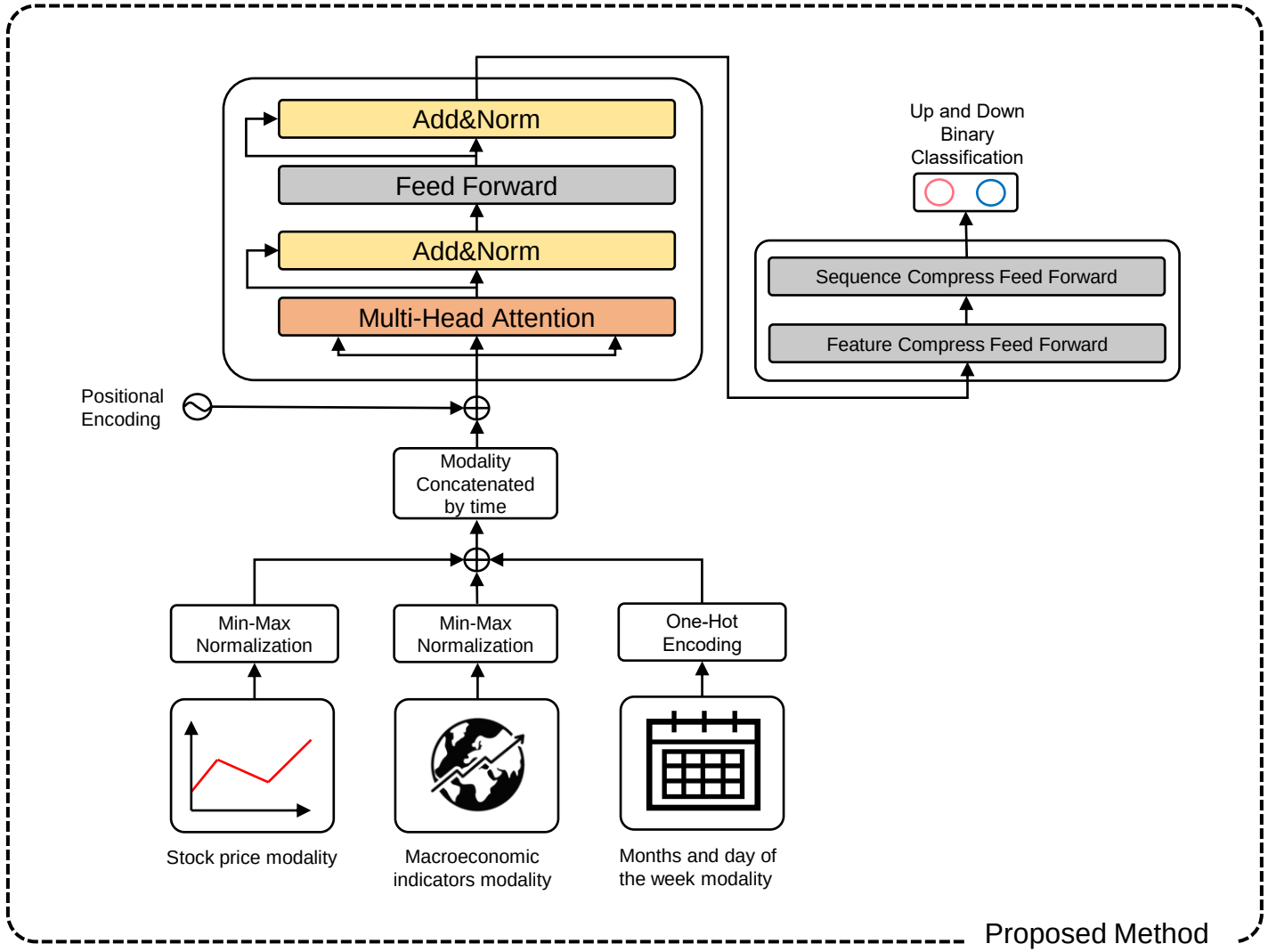
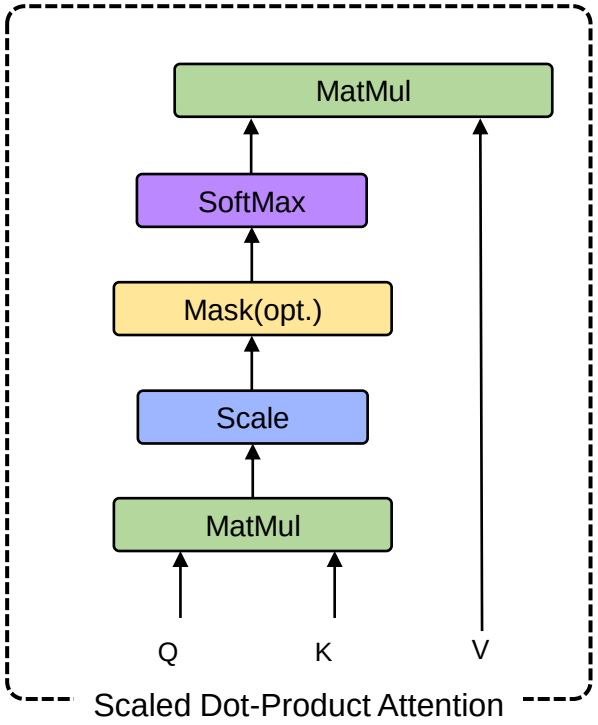


Chart pattern suggestion

Proposed Method

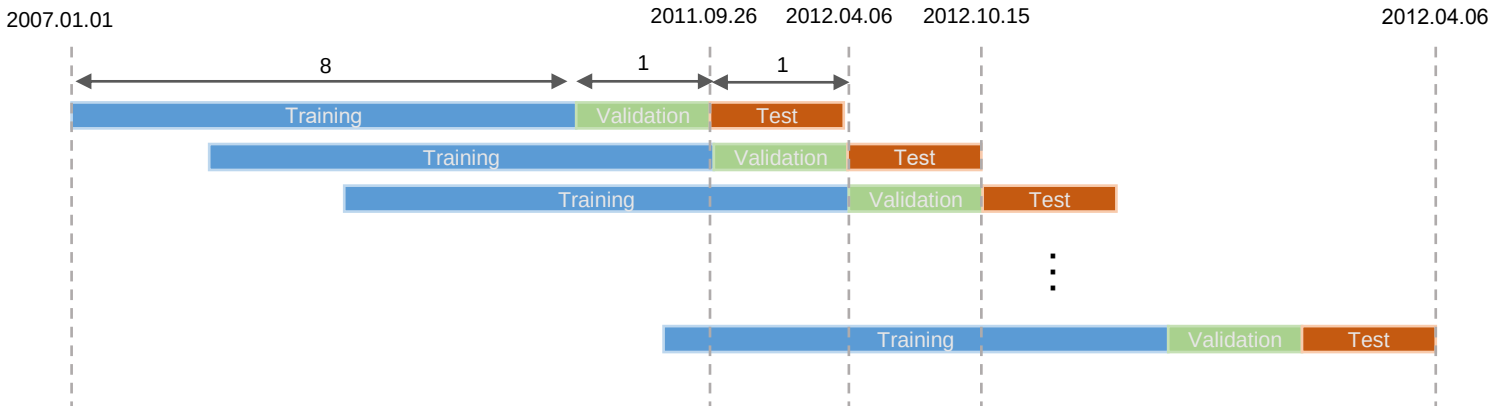
Multimodal early fusion Transformer





Experimental Setting

Time series blocked cross-validation



[1] W. Lu, J. Li, J. Wang, and L. Qin, “A cnn-bilstm-am method for stock price prediction,” Neural Computing and Applications, vol. 33, no. 10, pp. 4741–4753, 2021.

Experimental Setting

Comparison algorithm

- CNN-BiLSTM-AM [1]: Hybrid deep learning model
- AdvLSTM [2]: Adversarial Training
- DTML [3]: learning correlation between stocks # Transformer

Performance Measure

$$Accuracy(\%) = \frac{TP + TN}{TP + TN + FP + FN} \times 100$$

[1] W. Lu, J. Li, J. Wang, and L. Qin, “A cnn-bilstm-am method for stock price prediction,” Neural Computing and Applications, vol. 33, no. 10, pp. 4741–4753, 2021.

[2] J. Yoo, Y. Soun, Y.-c. Park, and U. Kang, “Accurate multivariate stock movement prediction via data-axis transformer with multi-level contexts,” in Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, pp. 2037–2045, 2021.

[3] F. Feng, H. Chen, X. He, J. Ding, M. Sun, and T.-S. Chua, “Enhancing stock movement prediction with adversarial training,” arXiv preprint arXiv:1810.09936, 2018.

Experimental Result

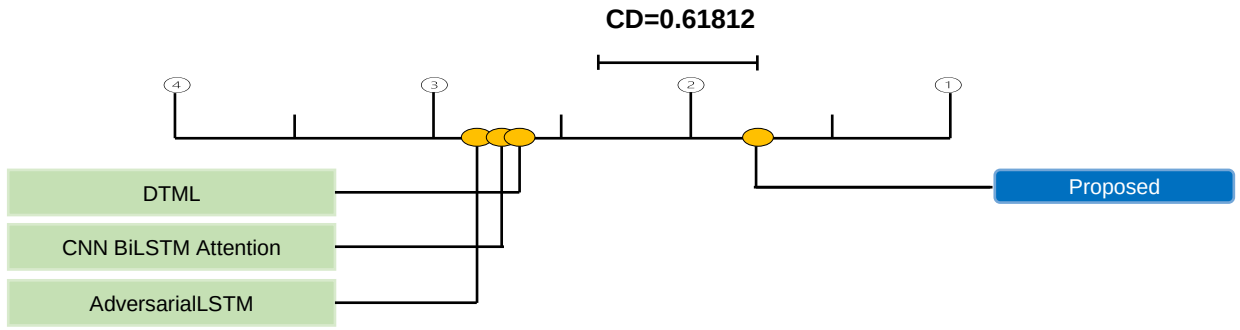
1. Shapiro-Wilk normality test [4]

Group	Shapiro-Wilk Statistic	Critical values (α = 0.05)
Proposed Method	0.9485	0.947
CNN-BiLSTM-AM	0.9791	
AdvLSTM	0.9725	
DTML	0.9727	

2. Friedman test (Multiple comparisons) [4]

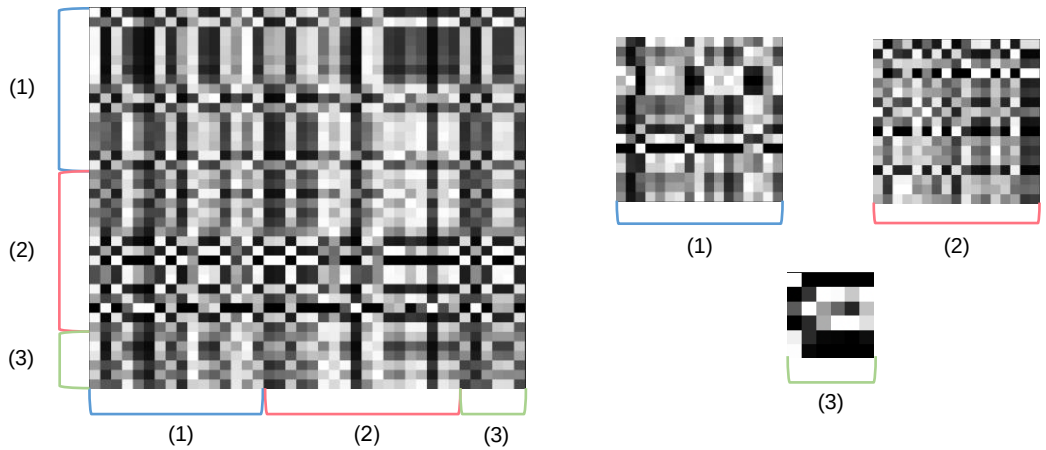
Evaluation measure	Friedman statistic	Critical values (α = 0.05)
Accuracy	24.084	2.557

3. Bonferroni-Dunn test (Post-hoc test) [4]



[4] B. Trawi´nski, M. Sm ˛etek, Z. Telec, and T. Lasota, “Nonparametric statistical analysis for multiple comparison of machine learning regression algorithms,” Internati onal Journal of Applied Mathematics and Computer Science, vol. 22, no. 4, pp. 867–881, 2012.

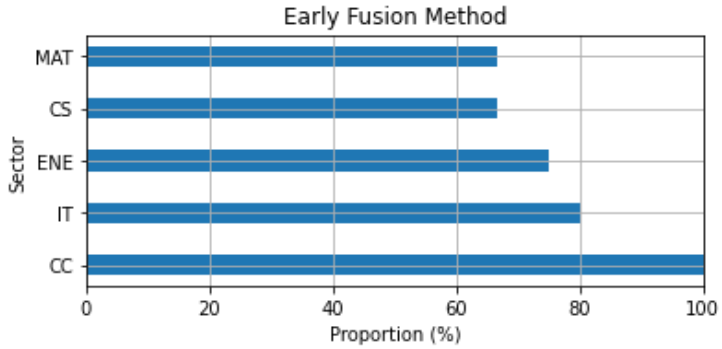
Analysis

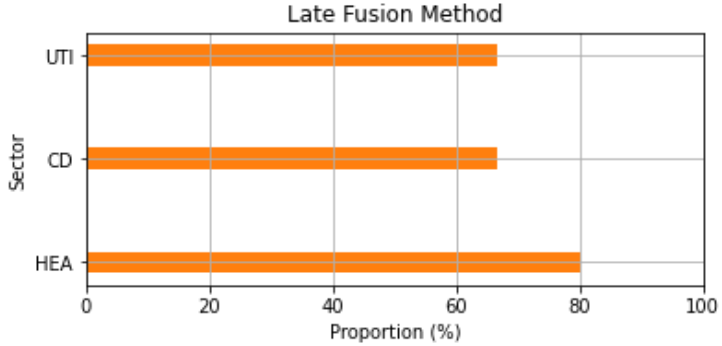


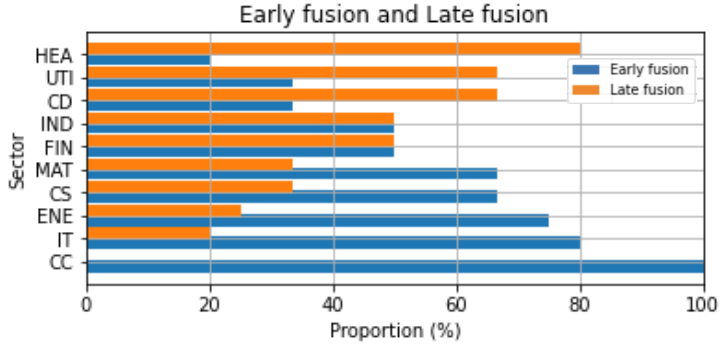
(a) Early fusion

(b) Late fusion

	The Number of data with high accuracy
Early fusion method	30
Late fusion method	20







To Do List

다음 주

- 졸업논문 완성

Thank you

Effective Music Genre Classification using Early Fusion Convolutional Neural Network

Sung-Hyun Cho

saintcho94@gmail.com

19th Oct, 2022

Contents

To Do List from Previous Seminar

- ✓ IEEE ACCESS 및 졸업 논문 작성
  - ➢ 논문 초안 작성
  - ➢ 불필요한 어휘/내용 삭제/수정
  - ➢ Model Architecture 추가
  - ➢ GMD-G 데이터셋 정확도 확인

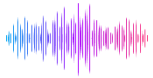
Music Information Retrieval (MIR)

- ✓ Concerned with the **extraction** and **inference of meaningful features** from music.
- ✓ Aims at **making the vast store of music** available to individuals.
- ✓ Different representations of **music-related subjects** and **items are considered**.

Typical MIR subfields

Music Genre Classification

Music Generation



Music Mood Classification

Playlist Generation

Urban Sound Classification

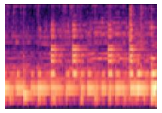
Audio Fingerprinting

⋮

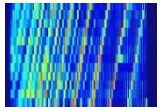
Feature Extraction

Visual Data

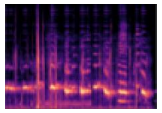
STFT



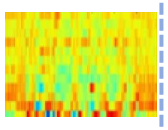
Chroma Features



Mel Spectrogram



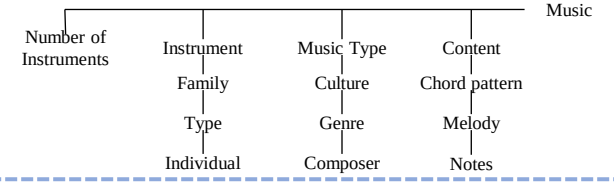
MFCC



Tabular Data

Title	Artist	Album	Track	Year	Length	Size	Last Modified	Filename
(Mama) He Treats Your Daughter Mean	Brown, Ruth	Best of Ruth Brown [Rhino]	8	1996		30 352,00 KB	09-09-2010	001.mp3
Night In Tunisia	Blakey, Art & the Jazz Messengers	Night In Tunisia [1960]	1	1960		30 348,00 KB	09-09-2010	002.mp3
Along Came Jones	Coasters	50 Coastin' Classics Anthology	27	1992		30 352,00 KB	09-09-2010	003.mp3
Back In Black	AC/DC	Back in Black	6	1980		30 352,00 KB	09-09-2010	004.mp3
Bad Reputation	Jett, Joan & Teh Blackhearts	Bad Reputation [Bonus Tracks]	1	1992		29 340,00 KB	09-09-2010	005.mp3
Boogie Woogie Stomp	Ammons, Albert	First Day	5	1939		30 352,00 KB	09-09-2010	006.mp3

Metadata

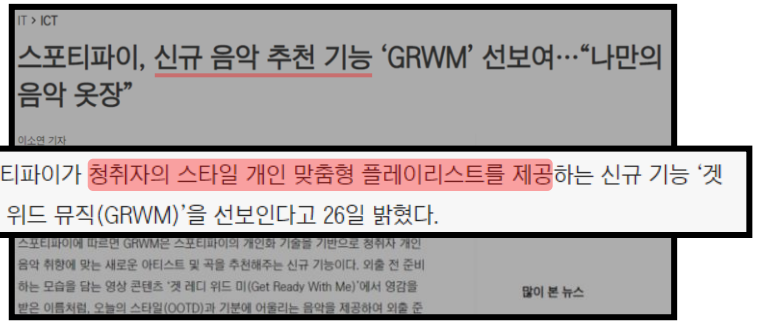


Classifiers

- K-Nearest Neighbor
- Naïve Bayes
- Support Vector Machine
- Decision Tree
- Random Forest
- Artificial Neural Network

⋮

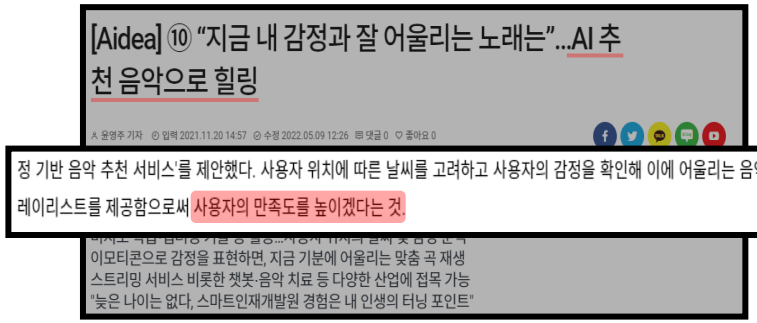
Results



IT > ICT

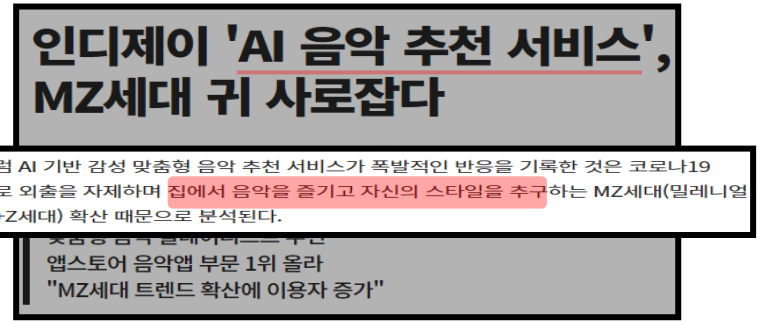
스포티파이, 신규 음악 추천 기능 'GRWM' 선보여…"나만의 음악 옷장"

스포티파이가 청취자의 스타일 개인 맞춤형 플레이리스트를 제공하는 신규 기능 '겟 레디 위드 뮤직(GRWM)'을 선보인다고 26일 밝혔다.



[Aidea] ⑩ "지금 내 감정과 잘 어울리는 노래는"...AI 추천 음악으로 힐링

정 기반 음악 추천 서비스'를 제안했다. 사용자 위치에 따른 날씨를 고려하고 사용자의 감정을 확인해 이에 어울리는 음악들의 플레이리스트를 제공함으로써 사용자의 만족도를 높이겠다는 것.



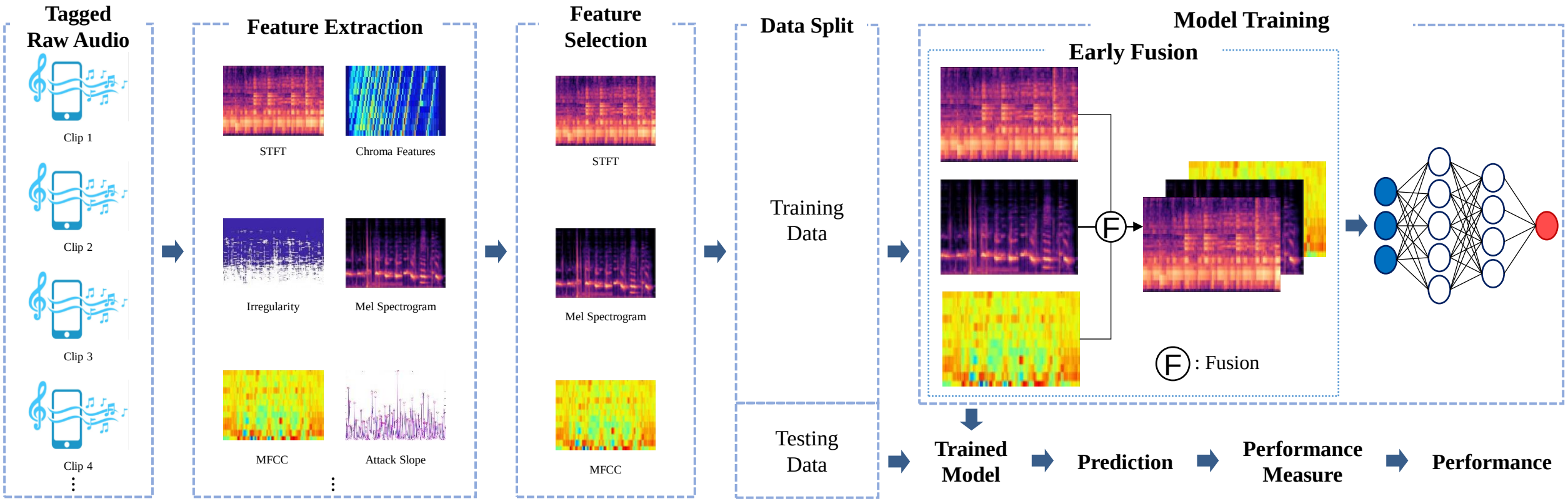
인디제이 'AI 음악 추천 서비스', MZ세대 귀 사로잡다

이처럼 AI 기반 감성 맞춤형 음악 추천 서비스가 폭발적인 반응을 기록한 것은 코로나19 등으로 외출을 자제하며 집에서 음악을 즐기고 자신의 스타일을 추구하는 MZ세대(밀레니얼 세대+Z세대) 확산 때문으로 분석된다.

앱스토어 음악앱 부문 1위 올라

"MZ세대 트렌드 확산에 이용자 증가"

Music Genre Classification using Convolutional Neural Network

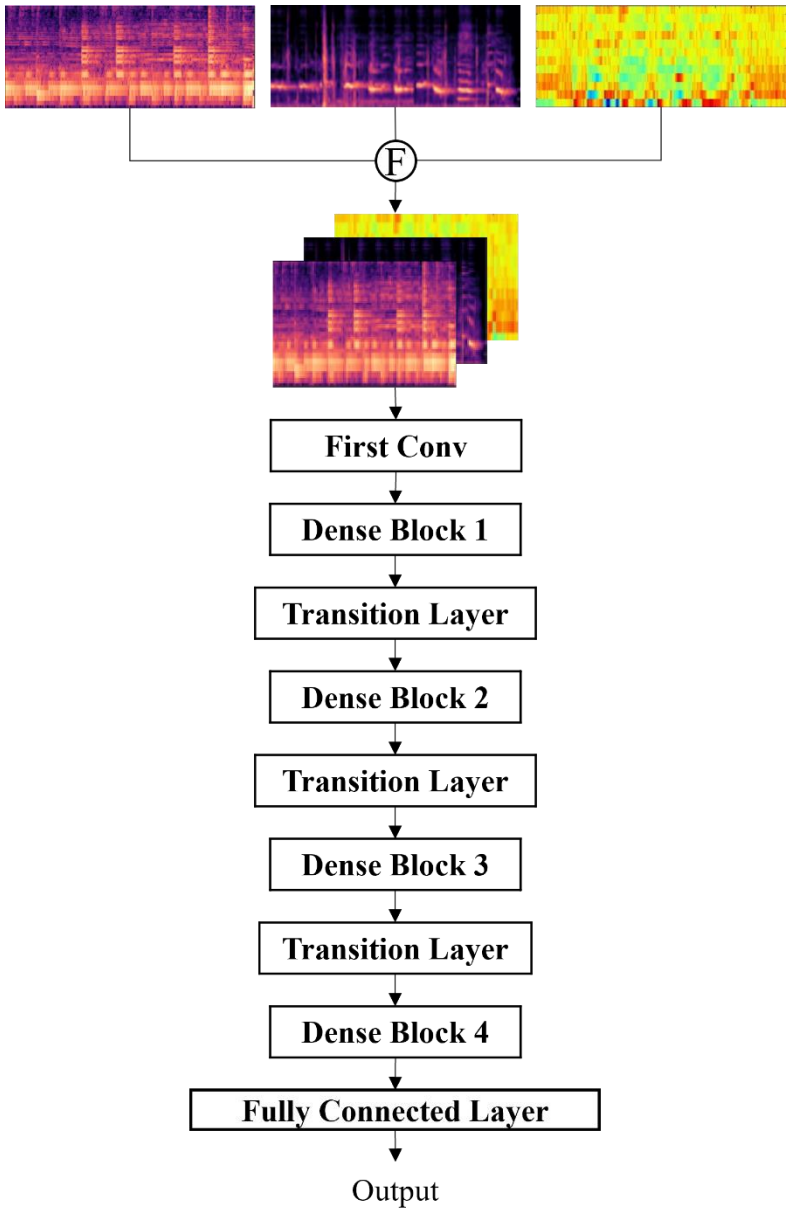


Proposed Method

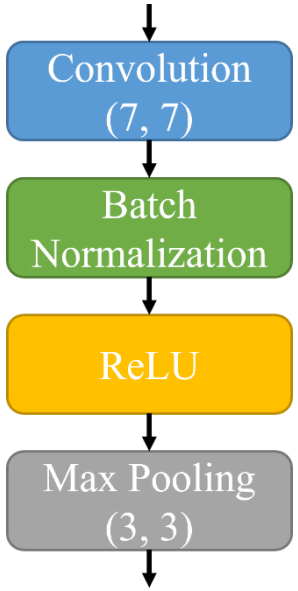
Model 그림

- ✓ Early Fusion 그림 디테일 강화하여 보완 (입력으로 들어가는 데이터 3개에 대한 예제 그림 추가).
- ✓ 전체 플로우를 보여주는 그림.
- ✓ Early Fusion과 Late Fusion 사이의 차이점을 보여주는 그림 추가
- ✓ Early Fusion과 Late Fusion의 러닝 커브(가장 차이가 두드러지는 1개 데이터)

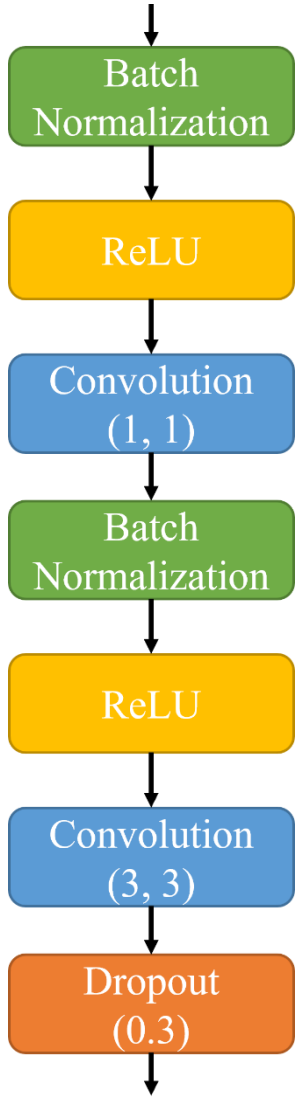
Proposed Method



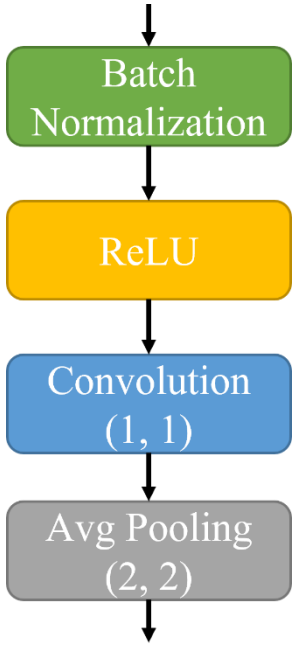
First Convolution



Dense Block (6/12/24/16)



Transition Layer



Proposed Method

Layers	Feature Map Size	Proposed Model
First Convolution	(3, 128, 628)	7×7 Convolution, stride = 2
	(3, 64, 324)	3×3 Max Pooling, stride = 2
Dense Block 1	(3, 64,324)	$\begin{bmatrix} 1 \times 1\ conv \\ 3 \times 3\ conv \end{bmatrix} \times 6$, (k = 32)
Transition Layer	(3, 64, 324)	1×1 Convolution
	(3, 32, 162)	2×2 Average Pooling, stride = 2
Dense Block 2	(3, 32, 162)	$\begin{bmatrix} 1 \times 1\ conv \\ 3 \times 3\ conv \end{bmatrix} \times 12$, (k = 32)
Transition Layer	(3, 32, 162)	1×1 Convolution
	(3, 16, 81)	2×2 Average Pooling, stride = 2
Dense Block 3	(3, 16, 81)	$\begin{bmatrix} 1 \times 1\ conv \\ 3 \times 3\ conv \end{bmatrix} \times 24$, (k = 32)
Transition Layer	(3, 16, 81)	1×1 Convolution
	(3, 8, 40)	2×2 Average Pooling, stride = 2
Dense Block 4	(3, 8, 40)	$\begin{bmatrix} 1 \times 1\ conv \\ 3 \times 3\ conv \end{bmatrix} \times 16$, (k = 32)
Classification Layer	(3, 1, 1)	7×7 Global Average Pooling
		1000D Fully-Connected, Softmax

Growth_rate (k) = 32

Proposed Method

		Layers	Input Number of Channels	Output Number of Channels	Number of Trainable Parameters	Output Image Shape
First Convolution		7x7 Conv	3	60	2,940	[3, 128, 648]
		3x3 MaxPool	-	-	-	[3, 64, 324]
Dense Block 1	**Dense Layer 1**	1x1 Conv	60	128	7,680	[3, 64, 324]
		3x3 Conv	128	32	36,864	[3, 64, 324]
	Dense Layer 2	1x1 Conv	92	128	11,776	[3, 64, 324]
		3x3 Conv	128	32	36,864	[3, 64, 324]
	Dense Layer 3	1x1 Conv	124	128	15,872	[3, 64, 324]
		3x3 Conv	128	32	36,864	[3, 64, 324]
	Dense Layer 4	1x1 Conv	156	128	19,968	[3, 64, 324]
		3x3 Conv	128	32	36,864	[3, 64, 324]
	Dense Layer 5	1x1 Conv	188	128	24,064	[3, 64, 324]
		3x3 Conv	128	32	36,864	[3, 64, 324]
	Dense Layer 6	1x1 Conv	220	128	28,160	[3, 64, 324]
		1x1 Conv	128	32	36,864	[3, 64, 324]
Transition Layer		1x1 Conv	252	126	31,752	[3, 64, 324]
		2x2 AvgPool	-	-	-	[3, 32, 162]

Number of Trainable Parameters = (Input Number of Channels) × (Kernel Size) × (Output Number of Channels)
1x1 conv 60x1x1x128 = 7680
3x3 conv 128x3x3x32 = 36864

Proposed Method

The number of input feature maps of l^{th}layer

- $k_0 + k \times (l-1) \quad (k_0 = 60, k = 32)$

Layer 1 $= 60 + 32 \times (1-1) = 60$
Layer 2 $= 60 + 32 \times (2-1) = 60 + 32 = 92$
Layer 3 $= 60 + 32 \times (3-1) = 60 + 64 = 124$
$\vdots$

Conv2d

$$H_{out} = \left\lfloor \frac{H_{in} + 2 \times padding[0] - dilation[0] \times (kernel\_size[0] - 1) - 1}{stride[0]} + 1 \right\rfloor$$

$$W_{out} = \left\lfloor \frac{W_{in} + 2 \times padding[1] - dilation[1] \times (kernel\_size[1] - 1) - 1}{stride[1]} + 1 \right\rfloor$$

AvgPool2d

$$H_{out} = \left\lfloor \frac{H_{in} + 2 \times padding[0] - kernel\_size[0]}{stride[0]} + 1 \right\rfloor$$

$$W_{out} = \left\lfloor \frac{W_{in} + 2 \times padding[1] - kernel\_size[1]}{stride[1]} + 1 \right\rfloor$$

MaxPool2d

$$H_{out} = \left\lfloor \frac{H_{in} + 2 \times padding[0] - dilation[0] \times (kernel\_size[0] - 1) - 1}{stride[0]} + 1 \right\rfloor$$

$$W_{out} = \left\lfloor \frac{W_{in} + 2 \times padding[1] - dilation[1] \times (kernel\_size[1] - 1) - 1}{stride[1]} + 1 \right\rfloor$$

Experimental Settings

Datasets

✓ Data Description

➢ **12 music genre datasets** from the creation stage and by post-processing

Dataset	Patterns	Avg. Length	# of Classes	Average Patterns per Class	±	Standard Deviation
Ballroom	698	30	8	87.250	±	17.555
Ballroom-Extended	4180	30	13	321.54	±	195.70
emoMusic-G	744	45	8	93.00	±	16.1776
Emotify-G	400	43	4	100.00	±	0.00
FMA-SMALL	7996	30	8	999.500	±	0.500
GiantStepsKey-G	604	80	23	24.9130	±	28.0274
GMD-G	1042	300	18	57.89	±	70.07
GTZAN	1000	30	10	100.00	±	0.00
HOMBURG	1886	10	9	209.56	±	134.87
ISMIR04	727	120	6	121.167	±	95.602
MICM	1137	360	7	162.429	±	119.717
Seyerlehner-Unique	3115	30	10	311.500	±	248.349

Experimental Settings

Hyperparameter Settings

Model	Batch Size	Epochs	Loss Function	Optimizer	LR	Weight Decay
Proposed	16	60	Cross Entropy Loss	Adam	1e-3	1e-4

Cross-Validation for Test Performance

- ✓ 10 iterations
- ✓ Data Split Ratio: (Train : Test) = (0.8 : 0.2)

Performance Measure

$$Accuracy(\%) = \frac{TP + TN}{TP + TN + FP + FN} \times 100$$

$TP : True\ Positive \quad TN : True\ Negative$

$FP : False\ Positive \quad FN : False\ Negative$

Evaluates the overall effectiveness of a model

➡ Higher value, Higher Performance

Experimental Settings

Comparison Algorithms

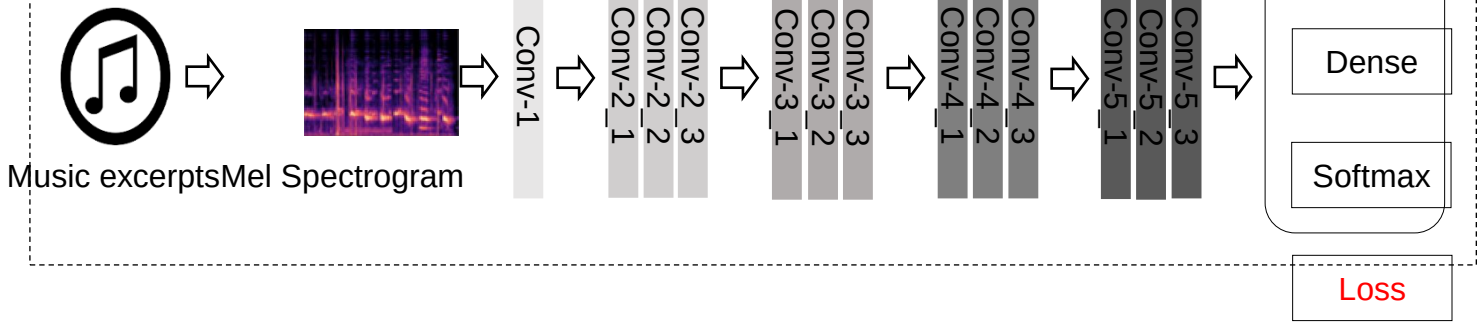
Model	Batch Size	Epochs	Loss Function	Optimizer	LR	Weight Decay
[1]	64	60	Proposed Loss Function	Adam	1e-3	1e-4
[2]	128		Cross Entropy Loss			
[3]	32		Cross Entropy Loss			
[4]	32		Cross Entropy Loss			

[1] Li, J., Han, L., Li, X., Zhu, J., Yuan, B., & Gou, Z. (2022). An evaluation of deep neural network models for music classification using spectrograms. Multimedia Tools and Applications, 81(4), 4621-4647.
[2] Pelchat, N., & Gelowitz, C. M. (2020). Neural network music genre classification. Canadian Journal of Electrical and Computer Engineering, 43(3), 170-173.
[3] Cheng, Y. H., Chang, P. C., & Kuo, C. N. (2020, November). Convolutional neural networks approach for music genre classification. In 2020 International Symposium on Computer, Consumer and Control (IS3C) (pp. 399-403). IEEE.
[4] Shah, M., Pujara, N., Mangaroliya, K., Gohil, L., Vyas, T., & Degadwala, S. (2022, March). Music Genre Classification using Deep Learning. In 2022 6th International Conference on Computing Methodologies and Communication (ICCMC) (pp. 974-978). IEEE.

Conventional methods of CNN-based MGC

Descriptions and Pros and Cons

[1]



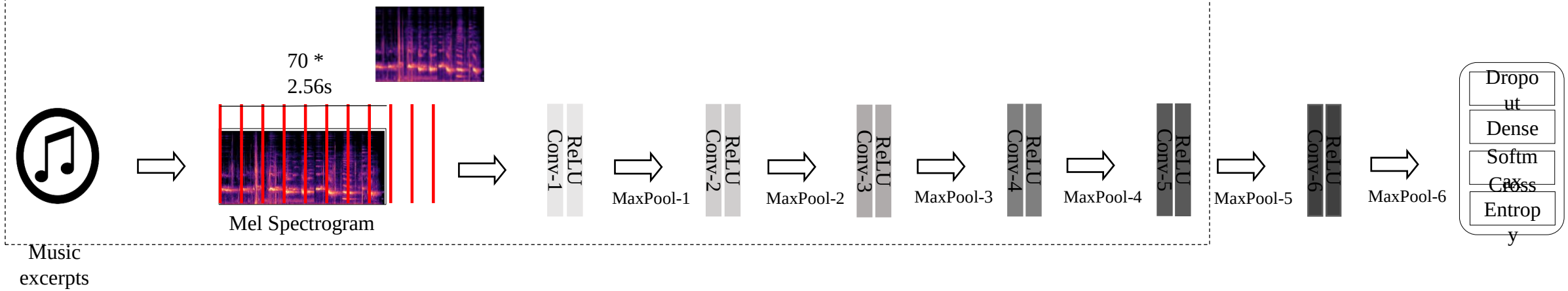
$$\text{Loss} = (1-t)\cdot Loss1 + t\cdot Loss2$$

$t = a^{-b}$ $(a: balance\ base, b: balance\ factor)$

$\text{Loss1} = \text{Cross\_Entropy}(y_{\text{true}}(\text{Onehot encoding}), y_{\text{pred}}(\text{Sof}$

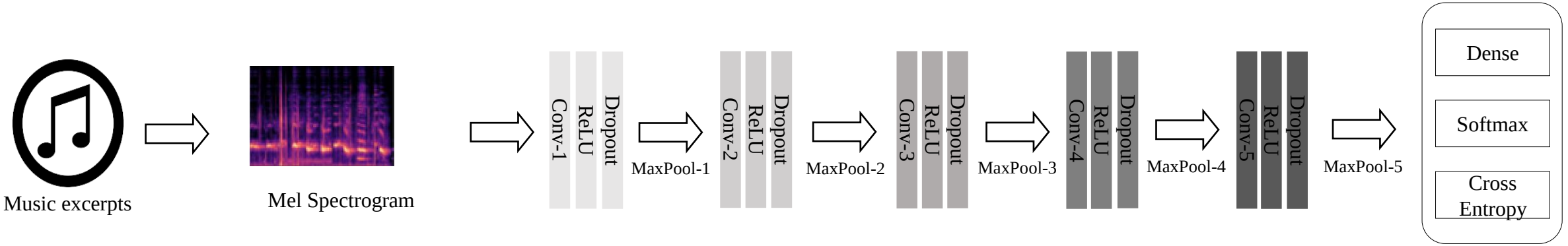
$\text{Loss2} = \text{Cross\_Entropy}(y_{\text{true}}(\text{Uniform Distribution functio}$

[2]

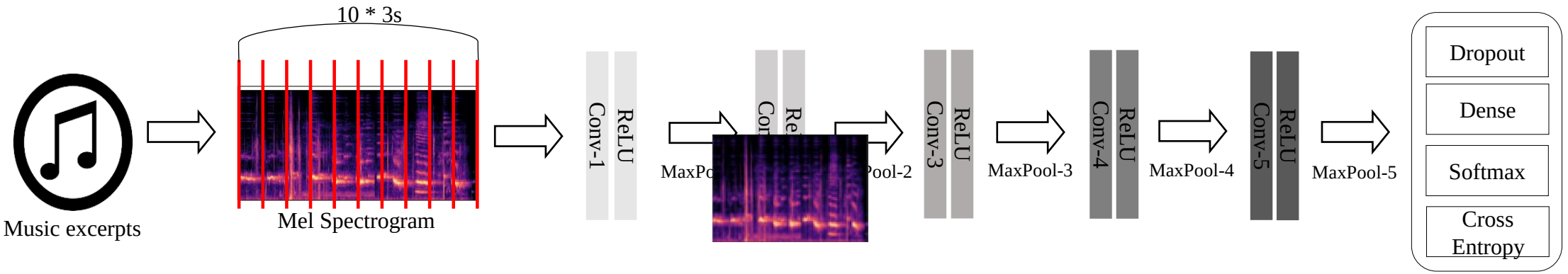


Conventional methods of CNN-based MGC

[3]



[4]



Experimental Results

Model Comparison Based on Accuracy

▼** : 99% significance level
▼* : 95% significance level

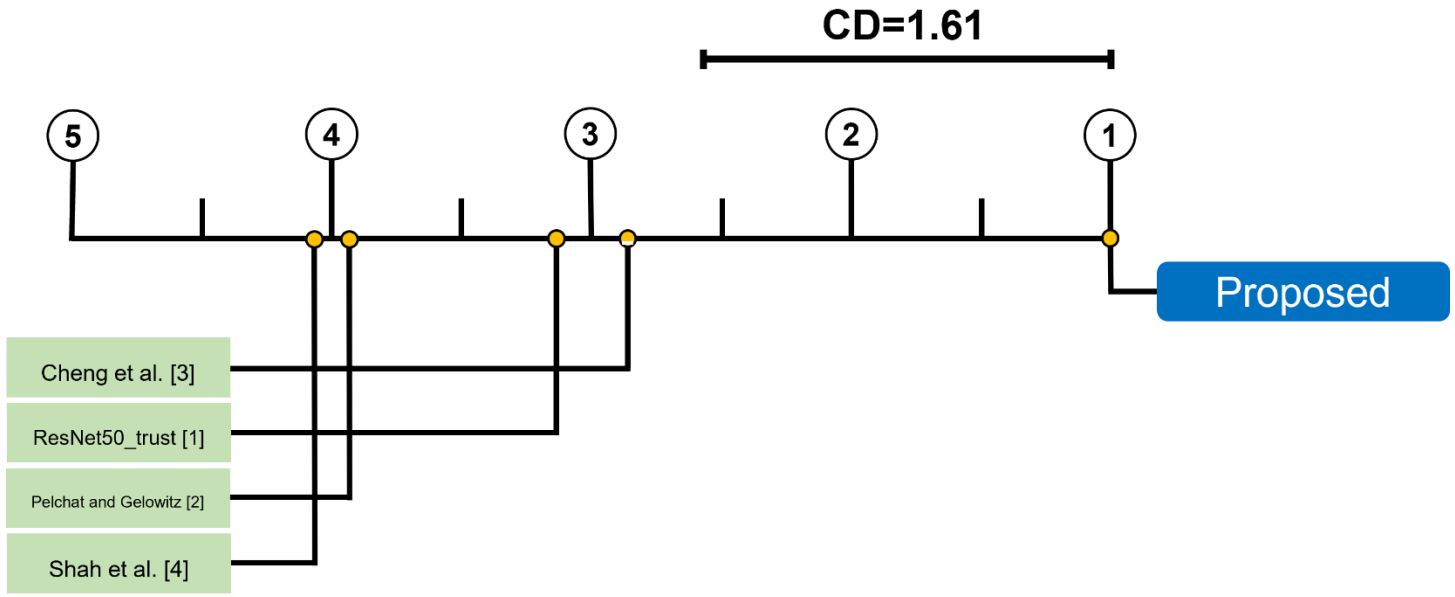
Dataset (Genre)	Proposed	ResNet50_trust	Pelchat and Gelowitz [2]	Cheng et al. [3]	Shah et al. [4]
Ballroom	$\mathbf{58.29 \pm 1.91}$	45.71 ± 2.86▼**	42.43 ± 2.29▼**	48.07 ± 4.79▼**	47.71 ± 4.58▼**
Ballroom-Extended	$\mathbf{85.49 \pm 1.06}$	61.82 ± 3.79▼**	44.68 ± 1.30▼**	62.88 ± 4.70▼**	54.38 ± 5.33▼**
emoMusic-G	$\mathbf{36.04 \pm 0.95}$	27.72 ± 3.50▼**	28.99 ± 1.78▼**	32.95 ± 2.92▼**	31.34 ± 3.66▼**
Emotify-G	$\mathbf{74.88 \pm 2.53}$	62.78 ± 2.70▼**	67.50 ± 4.17▼**	66.00 ± 3.32▼**	65.88 ± 5.62▼**
FMA-SMALL	$\mathbf{57.33 \pm 0.94}$	47.03 ± 1.31▼**	42.19 ± 1.16▼**	46.96 ± 1.62▼**	40.88 ± 1.73▼**
GiantStepsKey-G	$\mathbf{37.27 \pm 1.80}$	28.43 ± 3.15▼**	25.21 ± 2.53▼**	27.19 ± 2.90▼**	24.30 ± 2.11▼**
GMD-G	$\mathbf{67.13 \pm 1.13}$	61.91 ± 3.04▼**	63.44 ± 1.71▼**	39.38 ± 3.36▼**	35.36 ± 6.35▼**
GTZAN	$\mathbf{82.00 \pm 2.24}$	77.50 ± 3.81▼*	63.35 ± 2.08▼**	75.10 ± 3.84▼**	72.50 ± 2.13▼**
HOMBURG	$\mathbf{58.76 \pm 1.15}$	53.49 ± 1.59▼**	51.01 ± 1.51▼**	52.54 ± 2.55▼**	48.15 ± 2.14▼**
ISMIR04	$\mathbf{80.59 \pm 1.41}$	70.96 ± 2.93▼**	67.88 ± 1.70▼**	71.32 ± 1.37▼**	67.67 ± 1.99▼**
MICM	$\mathbf{43.25 \pm 2.32}$	37.81 ± 3.32▼**	39.30 ± 2.21▼**	38.42 ± 3.16▼**	39.34 ± 2.28▼**
Seyerlehner-Unique	$\mathbf{72.57 \pm 1.19}$	63.62 ± 2.56▼**	60.72 ± 2.73▼**	63.50 ± 2.79▼**	61.62 ± 1.70▼**
Total Win / Tie / Lose	48/0/0	16/16/16	3/15/30	15/19/14	5/16/27
Avg . Rank	**1.00**	3.17	3.92	2.83	4.08

Experimental Results

Friedman Test

Evaluation Measure	F_F	Critical Value($\alpha = 0.05$)
Accuracy	29.133	2.498

Bonferroni-Dunn Test



<Accuracy>

Conclusion

Conclusion

- ✓ Conducted a novel CNN model using early fusion.
- ✓ Achieve higher accuracy than conventional methods using a single input.

Contribution

- ✓ In international Publication (SCI(E))
  - ➢ Toward a Fair Evaluation and Analysis of Feature Selection for Music Tag Classification, ***IEEE Access,*** 2021, **Co-author**
- ✓ In International Conferences
  - ➢ Effective Music Genre Classification using Late Fusion Convolutional Neural Network with Multiple Spectral Features ***ICCE-Asia***, 2022, **1^{st} author**
  - ➢ An Efficient Neural Network based on Early Compression of Sparse CT Slice Images, ***PlatCon***, 2021, **Co-author**
- ✓ In Domestic Conferences
  - ➢ Music Genre Classification Using Multiple Musical Visual Features, ***IEIE***, 2022 , **1^{st} author**
  - ➢ Music Genre Classification using CNN architecture with feature concatenation, ***KISS Autumn Conf***, 2021, **Co-author**

To Do List

졸업

- ✓ Proposed Method 그림 추가
- ✓ ~10월 18일(화), 졸업논문 초안 작성
- ✓ ~10월 25일(화), 졸업논문 완성
- ✓ ~10월 31일(월), 심사위원 추천서 제출
- ✓ 11월 2일(수)~11월 8일(화), 졸업논문 발표 (발표 1주일 전 졸업논문의 스프링 제본판 전달)
- ✓ 11월 16일(수), 졸업논문 심사보고서 제출 마감

ICCE-Asia

- ✓ 10월 28일(금) 09:00~10:20, 발표

To Do List

Patent

✓ P20220244 (**자동 음악 태그 분류 성능 향상을 위한 음악 특징 선택 시스템 및 방법**)

- 중앙대학교 시스템에 따르면, P2022024건과 관련된 논문이 21.10.28에 공개된 것으로 확인
- 발명의 공개에 따른 '공지예외' 주장을 위해 **22.10.28 이전에 출원 완료** 예정

Thank you

Appendix

Appendix: Comparison Between Early Fusion and Late Fusion

Model Comparison Based on Accuracy

▼ : 95% significance level

Dataset (Genre)	Early Fusion	Late Fusion
Ballroom	**58.29 ± 1.91**	55.86 ± 3.41
Ballroom-Extended	**85.49 ± 1.06**	70.35 ± 3.25▼
emoMusic-G	**36.04 ± 0.95**	35.77 ± 2.49▼
Emotify-G	**74.88 ± 2.53**	72.00 ± 4.05▼
FMA-SMALL	**57.33 ± 0.94**	55.07 ± 0.84▼
GiantStepsKey-G	**37.27 ± 1.80**	35.29 ± 2.67▼
GMD-G	**67.13 ± 1.13**	65.50 ± 3.55
GTZAN	**82.00 ± 2.24**	79.00 ± 2.94
HOMBURG	**58.76 ± 1.15**	55.53 ± 2.39▼
ISMIR04	**80.59 ± 1.41**	79.13 ± 2.39
MICM	**43.25 ± 2.32**	41.18 ± 2.94
Seyerlehner-Unique	**72.57 ± 1.19**	70.85 ± 2.36▼
Avg . Rank	**1.00**	2.00

Appendix: Overall Experiment of Combination of Three Features

Comparison to Single Features (GTZAN)

Iteration	All	STFT	Mel-Spectrogram	MFCC
1	**85.50%**	72.50%	76.50%	64.00%
2	**83.00%**	70.50%	79.00%	67.00%
3	**84.00%**	74.50%	78.50%	59.00%
4	**82.00%**	76.00%	80.50%	61.50%
5	**80.50%**	72.00%	76.00%	60.00%
6	**82.00%**	73.00%	76.50%	66.00%
7	**84.00%**	75.50%	74.00%	68.50%
8	**81.50%**	76.00%	80.50%	62.00%
9	79.00%	79.00%	**80.00%**	59.00%
10	**78.50%**	75.50%	78.00%	62.00%
Average Accuracy	**82.00%**	74.40%	77.95%	62.90%

Model Summary

	First Convolution		Dense Block 1												Transition Layer	
			Dense Layer 1		Dense Layer 2		Dense Layer 3		Dense Layer 4		Dense Layer 5		Dense Layer 6			
Layers	7x7 Conv	3x3 MaxPool	1x1 Conv	3x3 Conv	1x1 Conv	3x3 Conv	1x1 Conv	3x3 Conv	1x1 Conv	3x3 Conv	1x1 Conv	3x3 Conv	1x1 Conv	1x1 Conv	1x1 Conv	2x2 AvgPool
Input Number of Channels	3	-	60	128	92	128	124	128	156	128	188	128	220	128	252	-
Output Number of Channels	60	-	128	32	128	32	128	32	128	32	128	32	128	32	126	-
Number of Trainable Parameters	8,820	-	7,680	36,864	11,776	36,864	15,872	36,864	19,968	36,864	24,064	36,864	28,160	36,864	31,752	-
Output Image Shape	[3, 128, 648]	[3, 64, 324]	[3, 64, 324]	[3, 64, 324]	[3, 64, 324]	[3, 64, 324]	[3, 64, 324]	[3, 64, 324]	[3, 64, 324]	[3, 64, 324]	[3, 64, 324]	[3, 64, 324]	[3, 64, 324]	[3, 64, 324]	[3, 64, 324]	[3, 32, 162]

	Dense Block 2												Transition Layer	
	Dense Layer 1	Dense Layer 2	Dense Layer 3	Dense Layer 4	Dense Layer 5	Dense Layer 6	Dense Layer 7	Dense Layer 8	Dense Layer 9	Dense Layer 10	Dense Layer 11	Dense Layer 12		
Layers	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	2x2 AvgPool
Input Number of Channels	126	158	190	222	254	286	318	350	382	414	446	478	510	-
Output Number of Channels	128	128	128	128	128	128	128	128	128	128	128	128	255	-
Number of Trainable Parameters	16,128	20,224	24,320	28,416	32,512	36,608	40,704	44,800	48,896	52,992	57,088	61,184	130,050	-
Output Image Shape	[3, 32, 162]	[3, 32, 162]	[3, 32, 162]	[3, 32, 162]	[3, 32, 162]	[3, 32, 162]	[3, 32, 162]	[3, 32, 162]	[3, 32, 162]	[3, 32, 162]	[3, 32, 162]	[3, 32, 162]	[3, 32, 162]	[3, 16, 81]

Model Summary

	Dense Block 3																							
	Dense Layer 1	Dense Layer 2	Dense Layer 3	Dense Layer 4	Dense Layer 5	Dense Layer 6	Dense Layer 7	Dense Layer 8	Dense Layer 9	Dense Layer 10	Dense Layer 11	Dense Layer 12	Dense Layer 13	Dense Layer 14	Dense Layer 15	Dense Layer 16	Dense Layer 17	Dense Layer 18	Dense Layer 19	Dense Layer 20	Dense Layer 21	Dense Layer 22	Dense Layer 23	Dense Layer 24	Transition Layer	
Layers	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x2 Conv	1x2 Conv	1x1 Conv	1x3 Conv	1x3 Conv	1x1 Conv	1x4 Conv	1x4 Conv	1x1 Conv	1x2 Conv	1x2 Conv	1x1 Conv	1x1 Conv	2x2 AvgPool
Input Number of Channels	255	287	319	351	383	415	447	479	511	543	575	607	639	671	703	735	767	799	831	863	895	927	959	991	1023	
Output Number of Channels	128	128	128	128	128	128	128	128	128	128	128	128	128	128	128	128	128	128	128	128	128	128	128	128	511	-
Number of Trainable Parameters	32640	36736	40832	44928	49024	53120	57216	61312	65408	69504	73600	77696	81792	85888	89984	94080	98176	102272	106368	110464	114560	118656	122752	126848	522753	-
Output Image Shape	[3, 16, 81]	[3, 16, 81]	[3, 16, 81]	[3, 16, 81]	[3, 16, 81]	[3, 16, 81]	[3, 16, 81]	[3, 16, 81]	[3, 16, 81]	[3, 16, 81]	[3, 16, 81]	[3, 16, 81]	[3, 16, 81]	[3, 16, 81]	[3, 16, 81]	[3, 16, 81]	[3, 16, 81]	[3, 16, 81]	[3, 16, 81]	[3, 16, 81]	[3, 16, 81]	[3, 16, 81]	[3, 16, 81]	[3, 16, 81]	[3, 16, 81]	[3,8,40]

	Dense Block 4														
	Dense Layer 1	**Dense Layer 2**	**Dense Layer 3**	**Dense Layer 4**	**Dense Layer 5**	**Dense Layer 6**	**Dense Layer 7**	**Dense Layer 8**	**Dense Layer 9**	**Dense Layer 10**	**Dense Layer 11**	**Dense Layer 12**	**Dense Layer 13**	**Dense Layer 14**	**Dense Layer 15**	**Dense Layer 16**
Layers	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x1 Conv	1x2 Conv	1x2 Conv	1x1 Conv	1x3 Conv
Input Number of Channels	511	543	575	607	639	671	703	735	767	799	831	863	895	927	959	991
Output Number of Channels	128	128	128	128	128	128	128	128	128	128	128	128	128	128	128	128
Number of Trainable Parameters	65408	69504	73600	77696	81792	85888	89984	94080	98176	102272	106368	110464	114560	118656	122752	126848
Output Image Shape	[3,8,40]	[3,8,40]	[3,8,40]	[3,8,40]	[3,8,40]	[3,8,40]	[3,8,40]	[3,8,40]	[3,8,40]	[3,8,40]	[3,8,40]	[3,8,40]	[3,8,40]	[3,8,40]	[3,8,40]	[3,8,40]

Contextual Embeddings for Hypernym Discovery

Geonil Yun

rjsdlf45@gmail.com

19th Oct, 2022

Contents

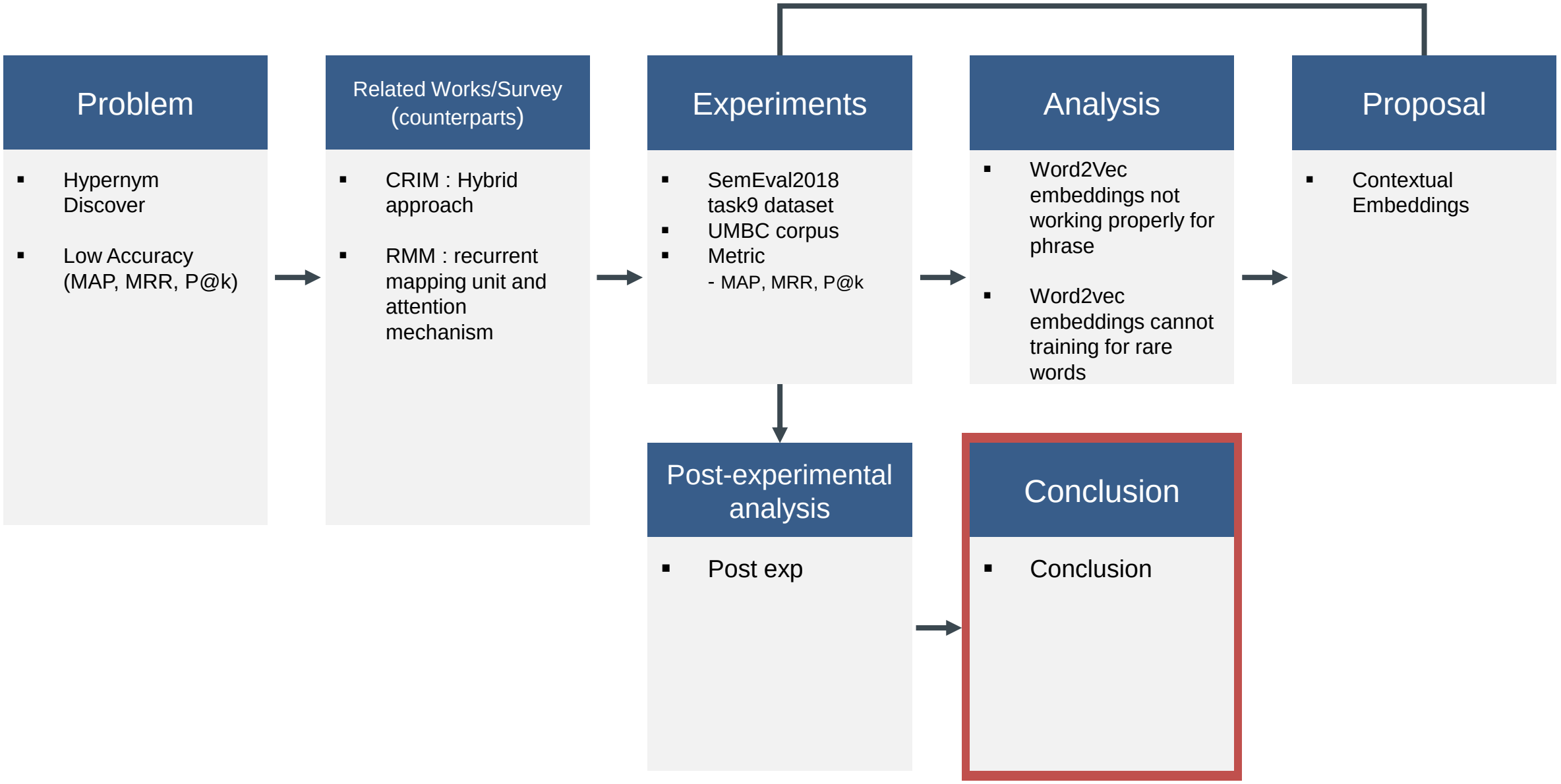
To Do List from Previous Seminar

- ✓ Attention mechanism -> Appendix로 이동

- ✓ Delete the attention mechanism from the proposed method.
  - ➢ Move to appendix

- ✓ Explain what vectors are good for hypernym discovery
  - ➢ Difference between contexture embedding and word2vec embedding

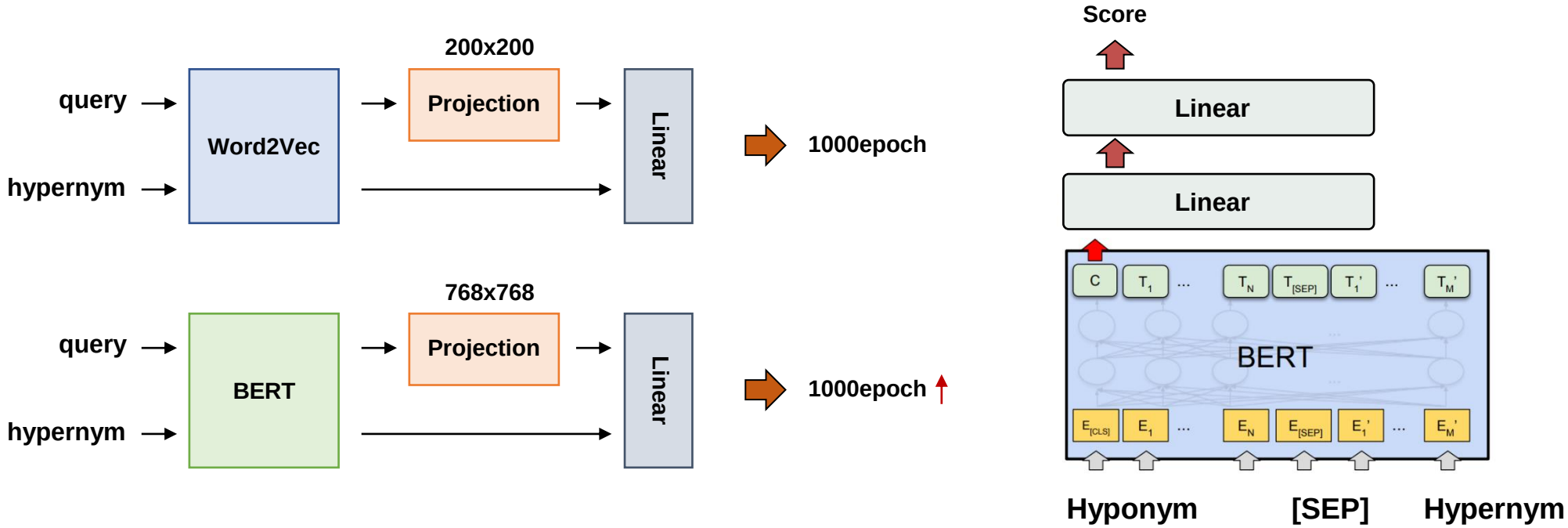
Overall Progress



Proposed Method

Contextual Embeddings for Projection Learning

- BERT embedding을 이용해 projection matrix로 사용하면 CRIM과 마찬가지로 projection matrix 학습에 오랜 시간이 필요함
- BERT embedding dimension이 더 크기 때문에 기존 CRIM보다 많은 학습을 요구함
- 이전 모델이 더 빠른 수렴을 보여줌 -> 이전 모델로 실험 진행중(5epoch 이하에서 수렴)

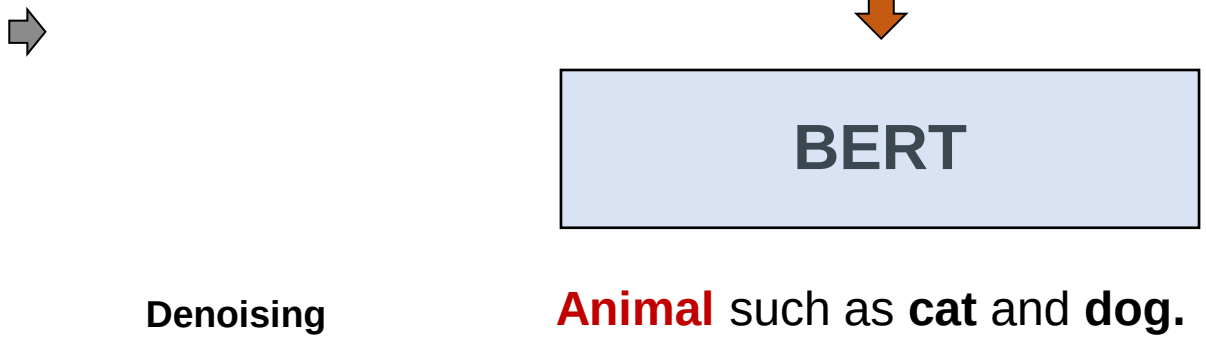


Proposed Method

Pretraining BERT for Hypernym Relation

- Hearst Pattern을 이용한 문장 추출
- Masking Language Modeling으로 BERT 사전학 습 후 기존 Proposed Method 학습

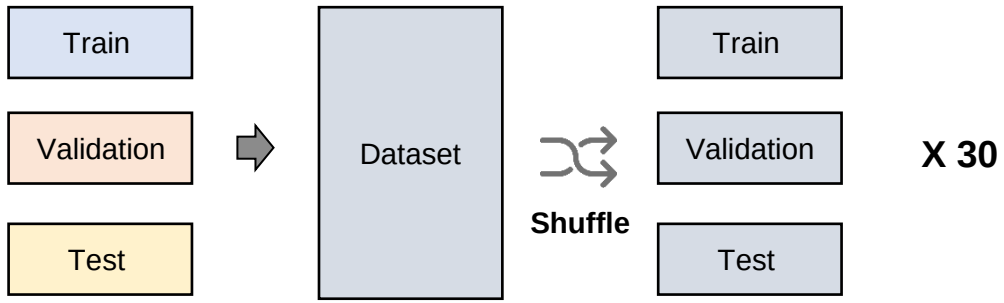
	Pattern
Hearst	Y such as (and|or) X
	such Y as (and|or) X
	X (and|or) other Y
	Y (including|especially) (and|or) X
Extended	Y (like|for example|which includes) (and|or) X
	Y other than (and|or) X
	not all Y are (and|or) X
	(and|or) X are (also|all) Y
	not Y so much as (and|or) X



Experimental Settings

Dataset

- The experiment was repeated 30 times with the shuffled dataset .
- We used 3 sub-tasks of SemEval2018 task9-Hypernym Discovery : 1A(general), 2A(Medical), 2B(Music)
- We employed 3 metrics for evaluation : MRR, MAP, P@k



Experimental Results

Dataset	Metric	Proposed	$CRIM_{r1}$	RMM
1A General	MRR		2.92 ± 1.90	0.45 ± 0.18
	MAP		1.68 ± 1.09	0.24 ± 0.10
	P@1		1.69 ± 1.46	0.27 ± 0.17
	P@3		1.34 ± 0.94	0.19 ± 0.10
	P@5		1.40 ± 0.95	0.21 ± 0.10
	P@15		2.12 ± 1.30	0.30 ± 0.11
2A Medical	MRR		3.06 ± 1.01	0.40 ± 0.41
	MAP		2.82 ± 0.79	0.17 ± 0.20
	P@1		0.88 ± 0.85	0.16 ± 0.20
	P@3		2.02 ± 0.70	0.19 ± 0.24
	P@5		2.57 ± 0.72	0.17 ± 0.20
	P@15		3.69 ± 1.02	0.18 ± 0.20
2B Music	MRR		22.77 ± 9.45	3.88 ± 1.46
	MAP		17.54 ± 6.40	2.90 ± 1.06
	P@1		16.16 ± 8.08	2.05 ± 1.38
	P@3		16.65 ± 6.68	2.50 ± 0.92
	P@5		17.05 ± 6.49	2.73 ± 0.91
	P@15		19.05 ± 6.25	3.41 ± 1.19

To Do List

- ✓ Experiments – repeat 30 times
  - ➢ 1A, 2A, 2B for Proposed model
  - ➢ it will take some time

- ✓ Explain what vectors are good for hypernym discovery
  - ➢ Difference between contexture embedding and word2vec embedding

- ✓ Pretraining은 우선 Appendix로 이동, Finetuning만으로 논문 마무리

Thank you

Restoration of Homoglyph Attack on Spam Using BERT Masked Language Model

Hanyong Lee

glhy0718@gmail.com

19th Oct, 2022

Contents

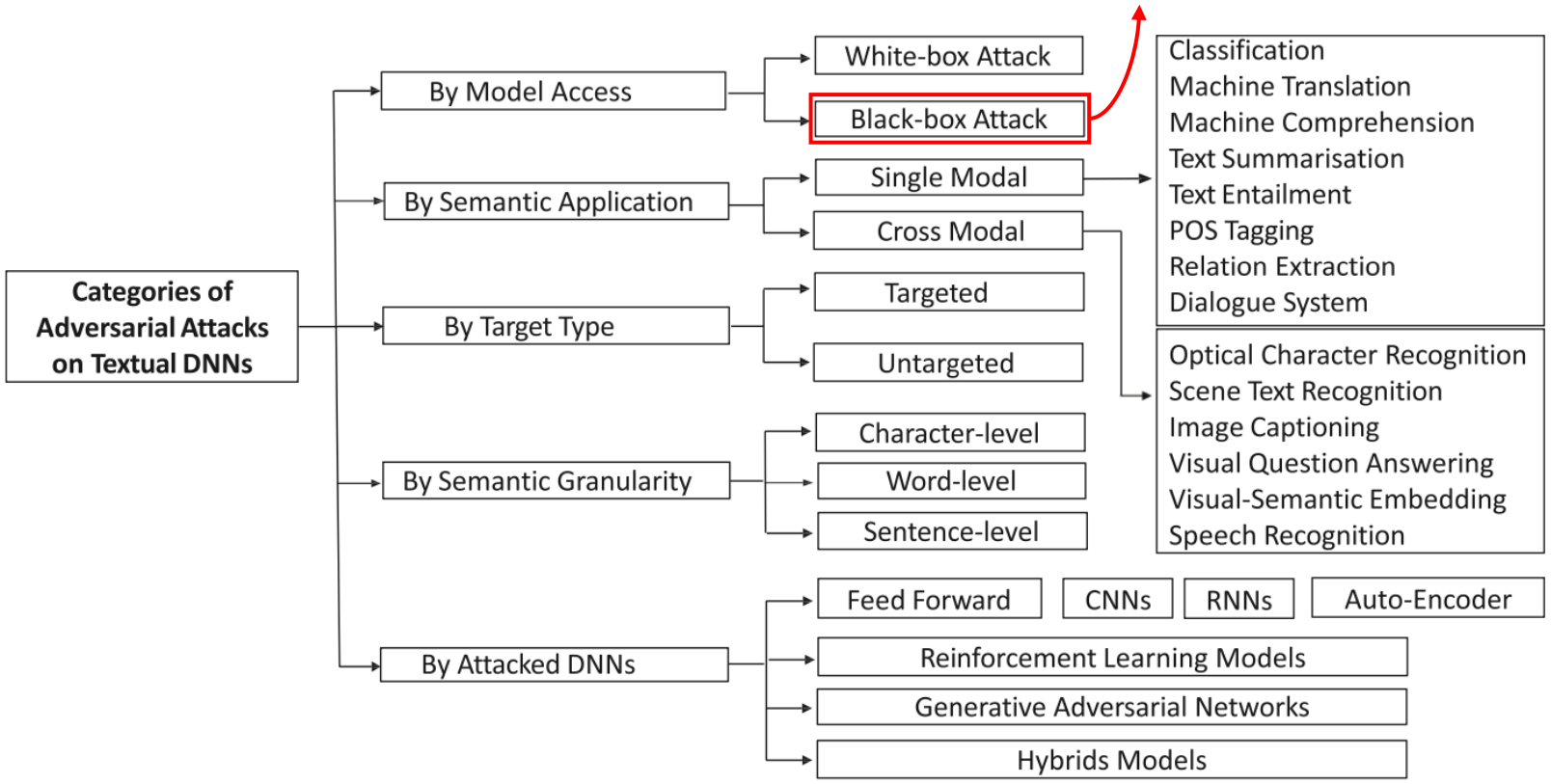
To-Do List from Previous Seminar

- ✓ 논문작성 시작 (Elsevier 양식 추천)
- ✓ PPT 수정
  - ➢ Introduction
    - Adversarial example on CV 제거
  - ➢ Experimental Settings
    - Dataset 표에 Pre-train/Finetuning dataset 표시
  - ➢ Experimental Results
    - Freidman / Dunn test 코드 재검토
    - Slide 21 표에서 proposed row 제거하고 Related Works Section으로 옮기기
  - ➢ Analysis
    - Proposed와 BERT 비교하는 내용 추가
      - output 차이, Proposed의 성능이 좋은 이유, BERT의 한계점 등의 관점에서
  - ➢ Conclusion
    - Contribution
      - Automated 관련 내용 제거
      - Char-BERT 제안에 대한 내용 추가

Introduction

NLP Attack

Adversarial examples in **Natural Language Processing**

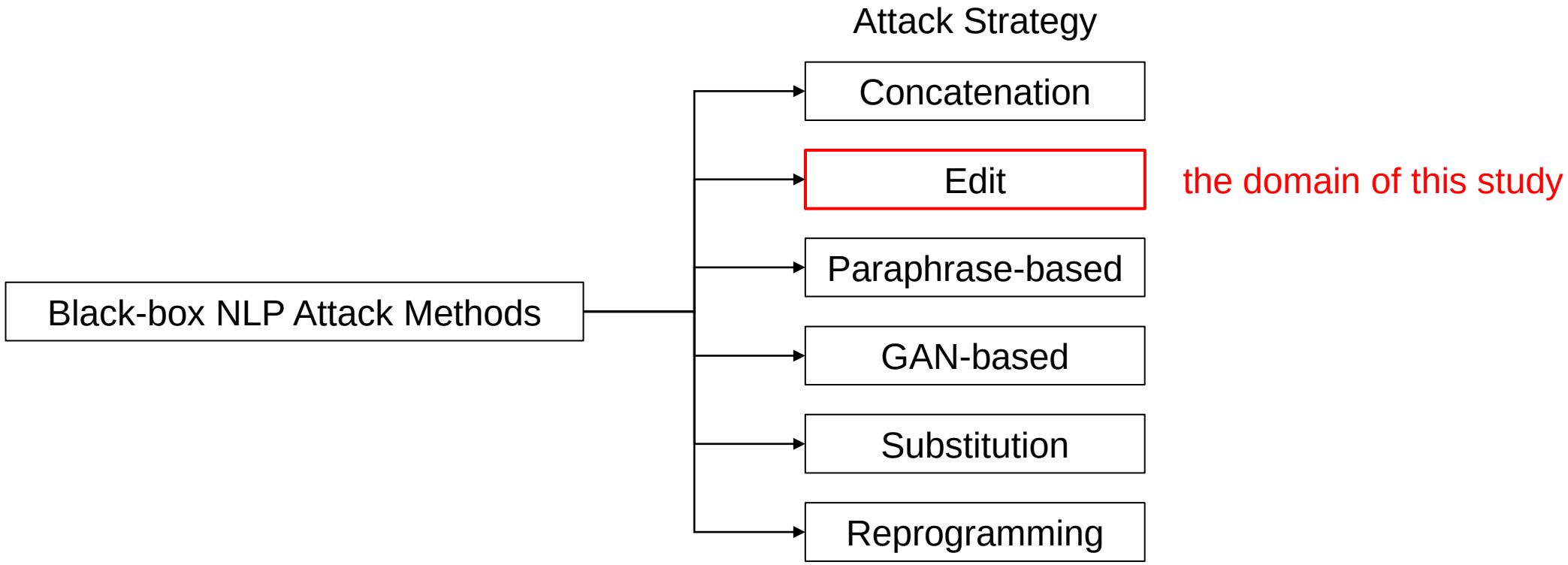


Zhang, W. E., Sheng, Q. Z., Alhazmi, A., & Li, C. (2020). Adversarial attacks on deep-learning models in natural language processing: A survey. ACM Transactions on Intelligent Systems and Technology (TIST), 11(3), 1-41.

Introduction

NLP Attack

Black-box NLP Attack Methods

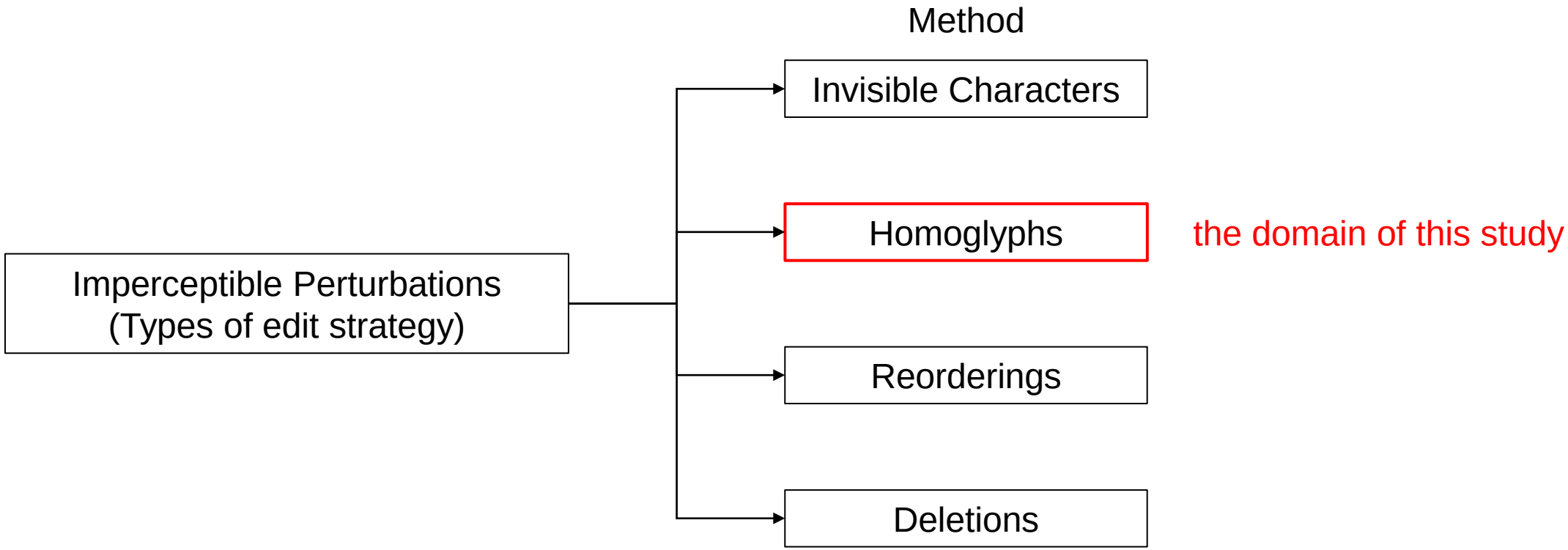


Zhang, W. E., Sheng, Q. Z., Alhazmi, A., & Li, C. (2020). Adversarial attacks on deep-learning models in natural language processing: A survey. ACM Transactions on Intelligent Systems and Technology (TIST), 11(3), 1-41.

Introduction

Homoglyph Attack

Imperceptible Perturbations

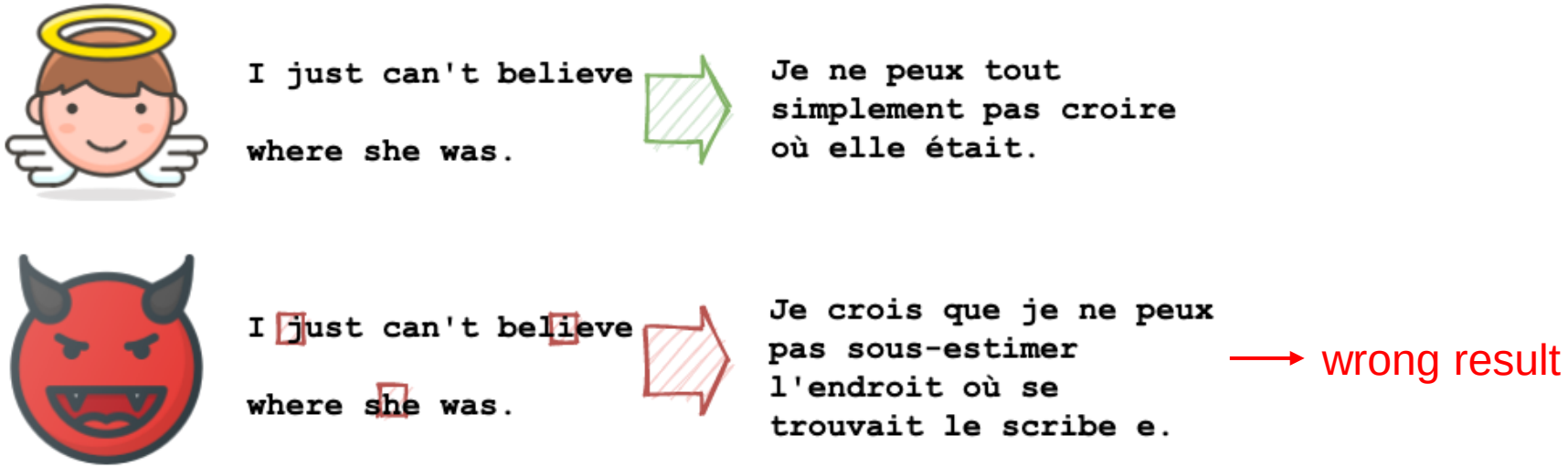


Boucher, N., Shumailov, I., Anderson, R., & Papernot, N. (2022, May). Bad characters: Imperceptible nlp attacks. In 2022 *IEEE Symposium on Security and Privacy (SP)* (pp. 1987-2004). IEEE.

Related Work

Homoglyph Attack

Example of an attack using homoglyph to confuse English-Franch translation task.

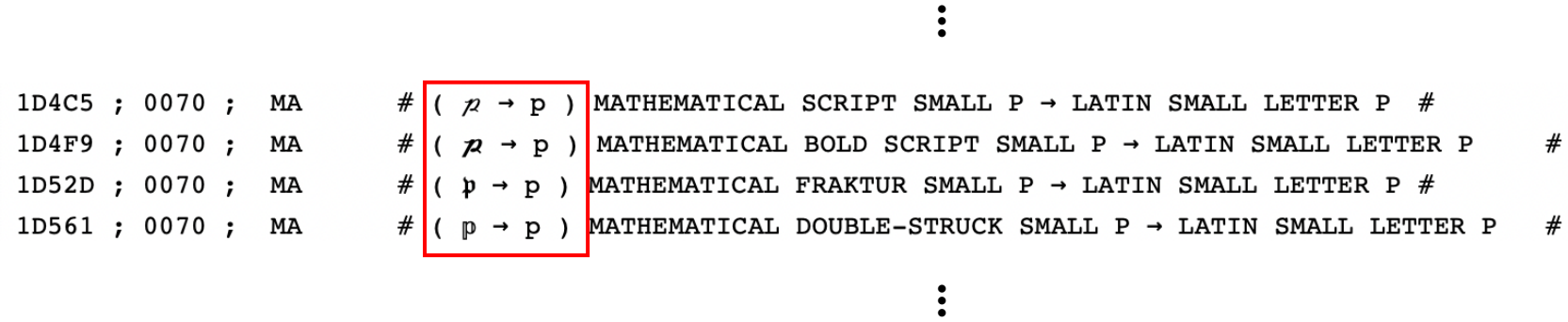


“j” is replaced with U+3F3(“j”), “i” with U+456(“i”), and “h” with U+4BB(“h”).

Boucher, N., Shumailov, I., Anderson, R., & Papernot, N. (2022, May). Bad characters: Imperceptible nlp attacks. In 2022 *IEEE Symposium on Security and Privacy (SP)* (pp. 1987-2004). IEEE.

Defense Methods of Homoglyph Attack

Translate table for confusable characters



```
1D4C5 ; 0070 ;  MA      # ( 𝓅 → p ) MATHEMATICAL SCRIPT SMALL P → LATIN SMALL LETTER P  #
1D4F9 ; 0070 ;  MA      # ( 𝓹 → p ) MATHEMATICAL BOLD SCRIPT SMALL P → LATIN SMALL LETTER P    #
1D52D ; 0070 ;  MA      # ( 𝔭 → p ) MATHEMATICAL FRAKTUR SMALL P → LATIN SMALL LETTER P #
1D561 ; 0070 ;  MA      # ( 𝕡 → p ) MATHEMATICAL DOUBLE-STRUCK SMALL P → LATIN SMALL LETTER P   #
```

It doesn’t contain some pairs.

- μ → u
- $ → S
- 5 → S
- …

Unicode Consortium 2022, *Unicode Utilities: Confusables.* accessed 10 September 2022, <https://www.unicode.org/Public/security/15.0.0/confusables.txt>.

Related Work

Defense Methods of Homoglyph Attack

SimChar database

H. Suzuki et al. used the following formula to calculate the similarity of two bitmap images of characters.

$$\Delta = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} |I_1(i,j) - I_2(i,j)|$$

pixel of image I_1, I_2 at coordinate (i, j)

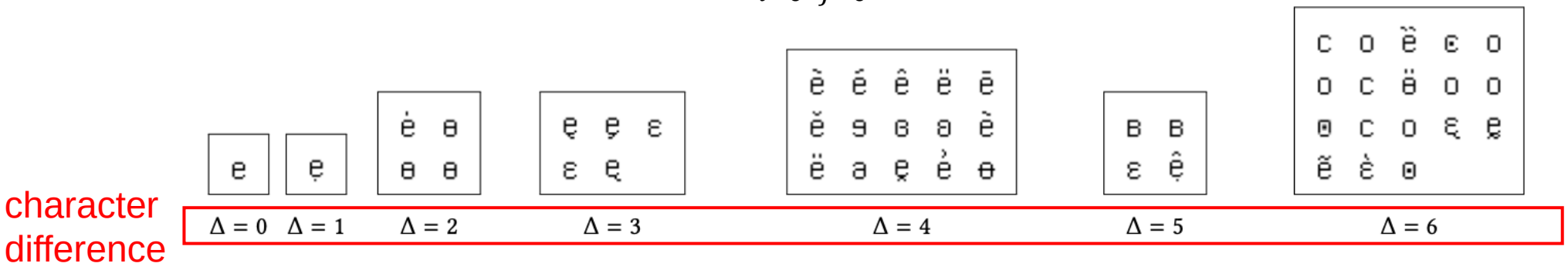


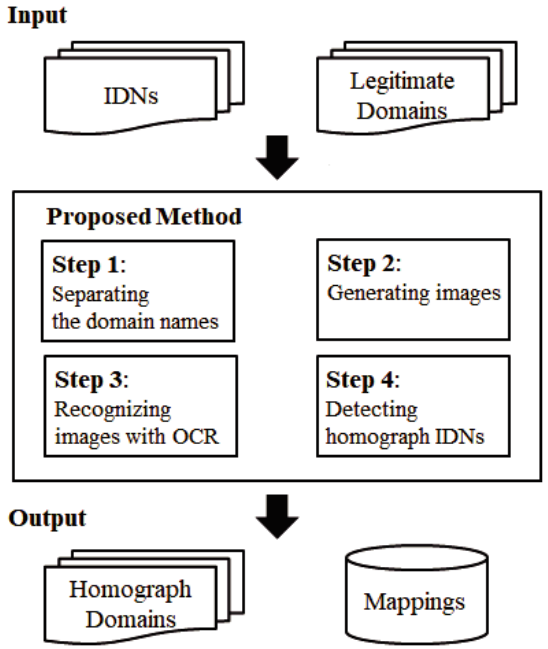
Figure 6: Basic Latin lowercase letter 'e' and characters under different values of the threshold Δ. In this work, characters with $\Delta \leq 4$ are detected as homoglyphs.

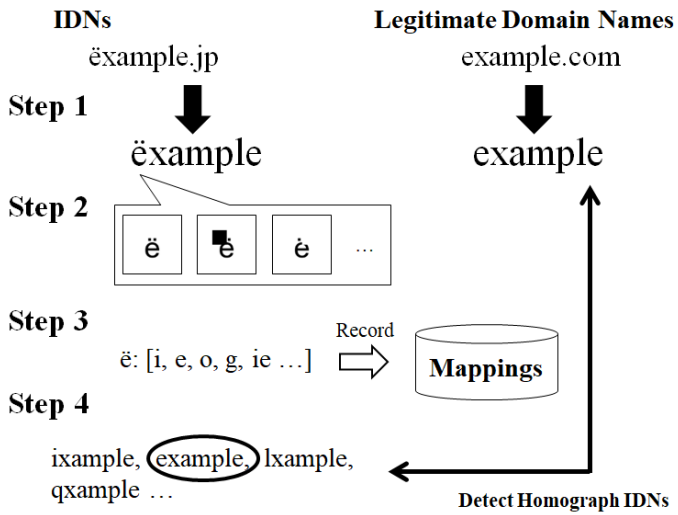
Suzuki, H., Chiba, D., Yoneya, Y., Mori, T., & Goto, S. (2019, October). ShamFinder: An automated framework for detecting IDN homographs. *In Proceedings of the Internet Measurement Conference* (pp. 449-462).

Defense Methods of Homoglyph Attack

OCR-based method

Sawabe et al. used the OCR-based Method to detect Homoglyph Attack.





Sawabe, Y., Chiba, D., Akiyama, M., & Goto, S. (2019). Detection method of homograph internationalized domain names with OCR. *Journal of Information Processing*, *27*, 536-544.

Defense Methods of Homoglyph Attack

Spellchecker-based method

Imam, N. H et al. used a spellchecker-based method to restore the homoglyph attacks.

Table 2
Some of the adversarial text examples used in experiments.

Original	Flipping	Swapping	Deletion
Busy	Bu$y	Bsuy	Bsy
Caller	C@1ler	Claler	Cller
Reply	Rep1y	Rpely	Rply
winner	w!nner	wniner	wnner

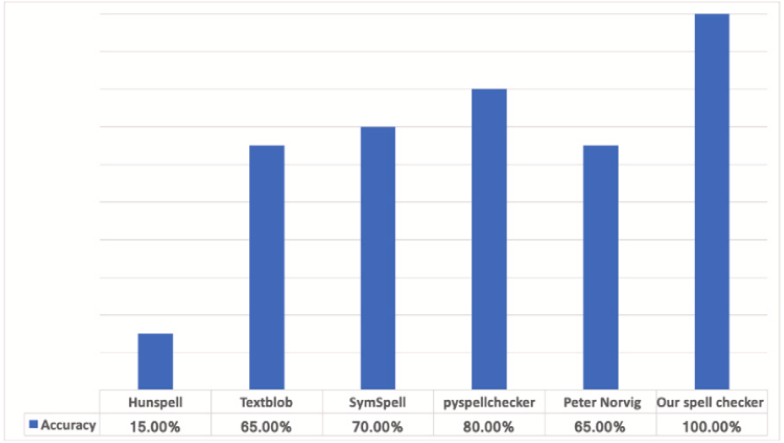


Fig. 9. Flipping.

Imam, N. H., Vassilakis, V. G., & Kolovos, D. (2022). OCR post-correction for detecting adversarial text images. *Journal of Information Security and Applications*, *66*, 103170.

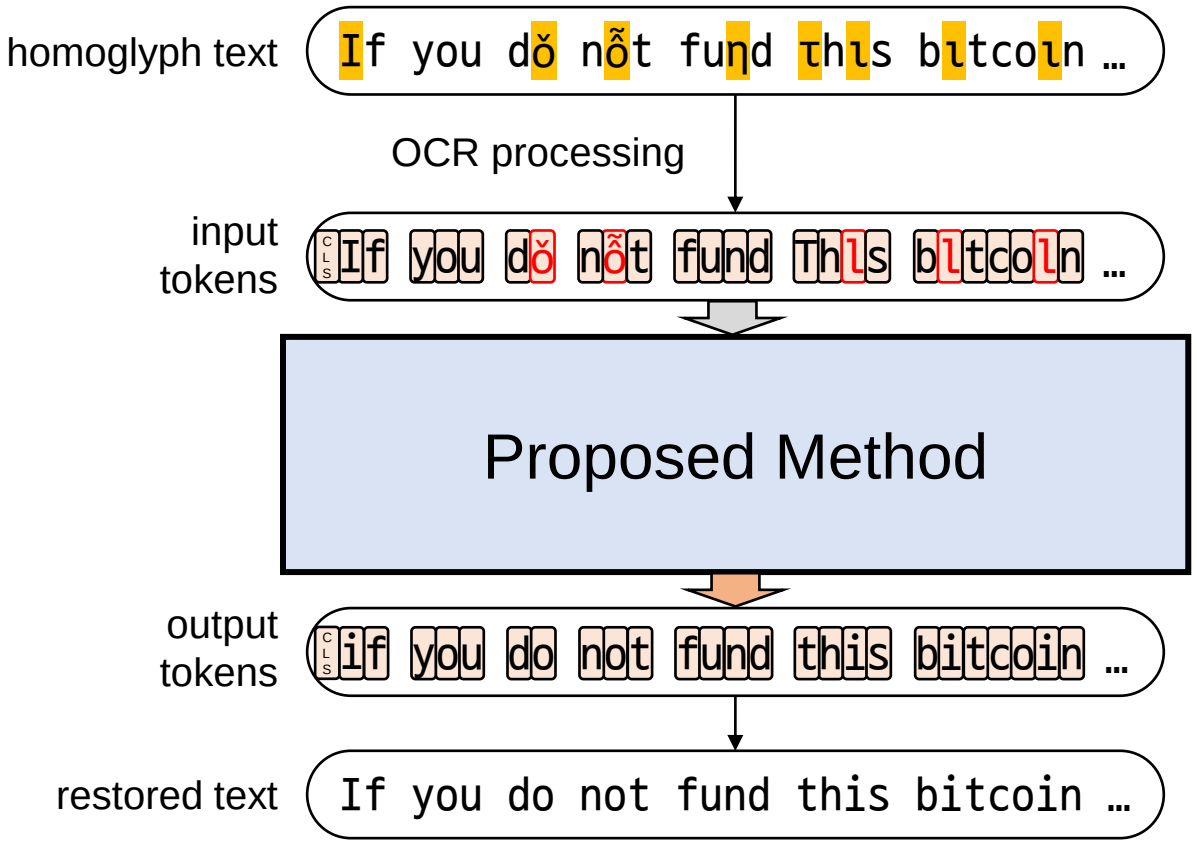
Related Work

Pros & Cons of related works

Algorithm	PROS	CONS
BERT	• Slightly robust on new homoglyphs	• Pre-raining the model takes a long time. • Word-based tokenization is not helpful for the task.
OCR	• Robust on new homoglyphs • Automated workflow	• OCR is not performing well on homoglyphs • Inability when the input & output lengths are different
SimChar DB	• Automated workflow on finding homoglyph pairs	• Not robust on new homoglyphs • Must manually add homoglyph pairs to the dictionary • Inability when the input & output lengths are different
Spell Checker	• Robust on new homoglyphs	• Must manually add words to the dictionary

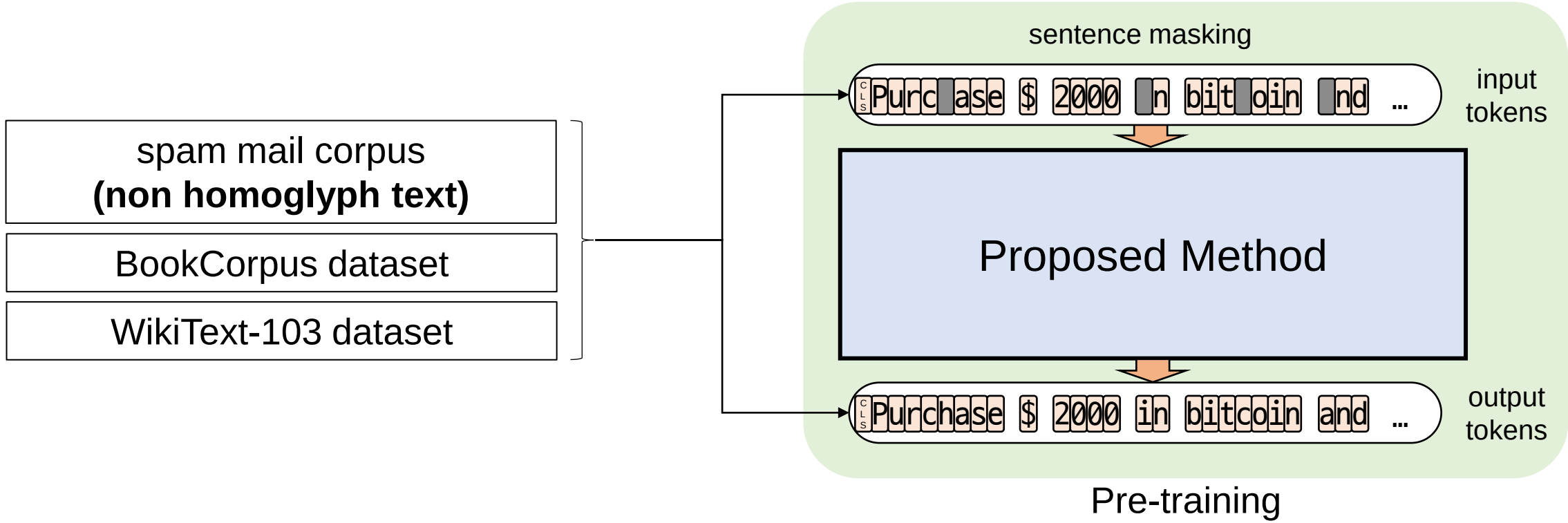
Proposed Method

Restoring Process Overview



Proposed Method

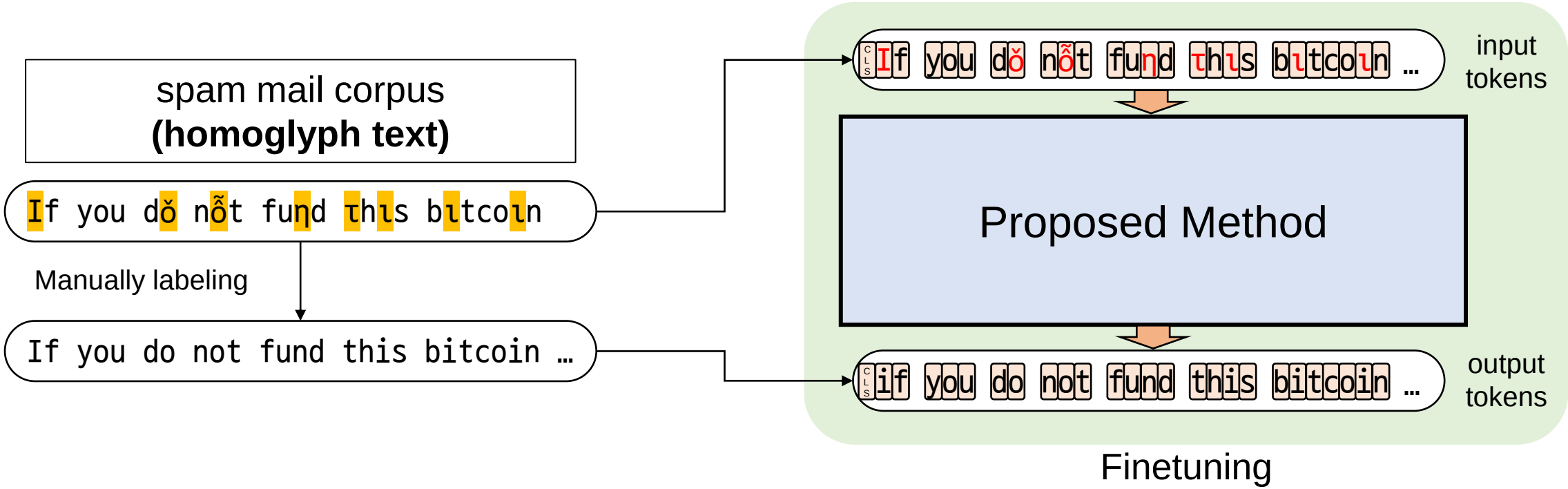
Pre-training



We pretrained the proposed model as a masked language model (MLM).

Proposed Method

Finetuning



Evaluated with a blind test (Train/Test set split ratio = 8:2 / Iteration = 50 times)

Experimental Settings

Dataset

		# of Examples		# of Tokens	
	Dataset	Train Set	Validation Set	Train Set	Validation Set
Pre-training dataset	WikiText-103	1,165,029	2,461	285,324,545	612,967
	BookCorpus (about 2% of the total dataset)	1,480,085	3,126	79,346,399	143,939
	bitcoinabuse.com dataset (non-homoglyph)	299,239	633	23,152,956	54,680
Finetuning dataset	bitcoinabuse.com dataset (homoglyph)	21,342	5,336	random split	random split

Experimental Settings

Model Parameters

Parameter	Value
Input Max Sequence	512
Vocab Size	881
# of Attention Heads	12
# of Hidden Layers	12
# of Model Parameters	86,720,369

Experimental Settings

Pre-training Setting

Parameter	Value
# of Epochs	25
Train / Validation split	refer the slide 14.
Used Datasets	• WikiText-103 • BookCorpus (about 2% of the total dataset) • bitcoinabuse.com dataset (non-homoglyph sentences)
Metric	Accuracy

Experimental Settings

Finetuning Setting

Parameter	Value
# of Epochs	5
Train / Validation split	8 : 2
# of Test Iteration	50
Used Datasets	• bitcoinabuse.com dataset (homoglyph sentences)
Metric	Accuracy

Experimental Results

Finetuned Model Evaluation - Accuracy

Algorithm	Word Accuracy
Proposed	**0.9959 ± 0.0008**
BERT	0.6713 ± 0.0029 ▼
OCR	0.6518 ± 0.0028 ▼
SimChar DB	0.5512 ± 0.0046 ▼
Spell Checker	0.9087 ± 0.0011 ▼

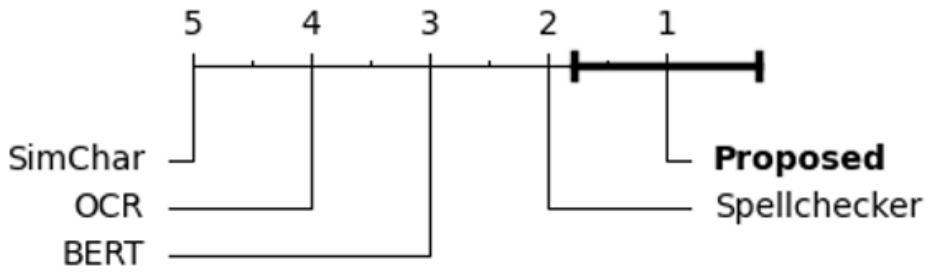
Word level accuracy and T-test with the accuracy of the proposed method
▼ means that the algorithm has a significantly lower performance than the proposed algorithm based on the t-test.
▼: $p < 0.01$

Experimental Results

Finetuned Model Evaluation – Friedman test

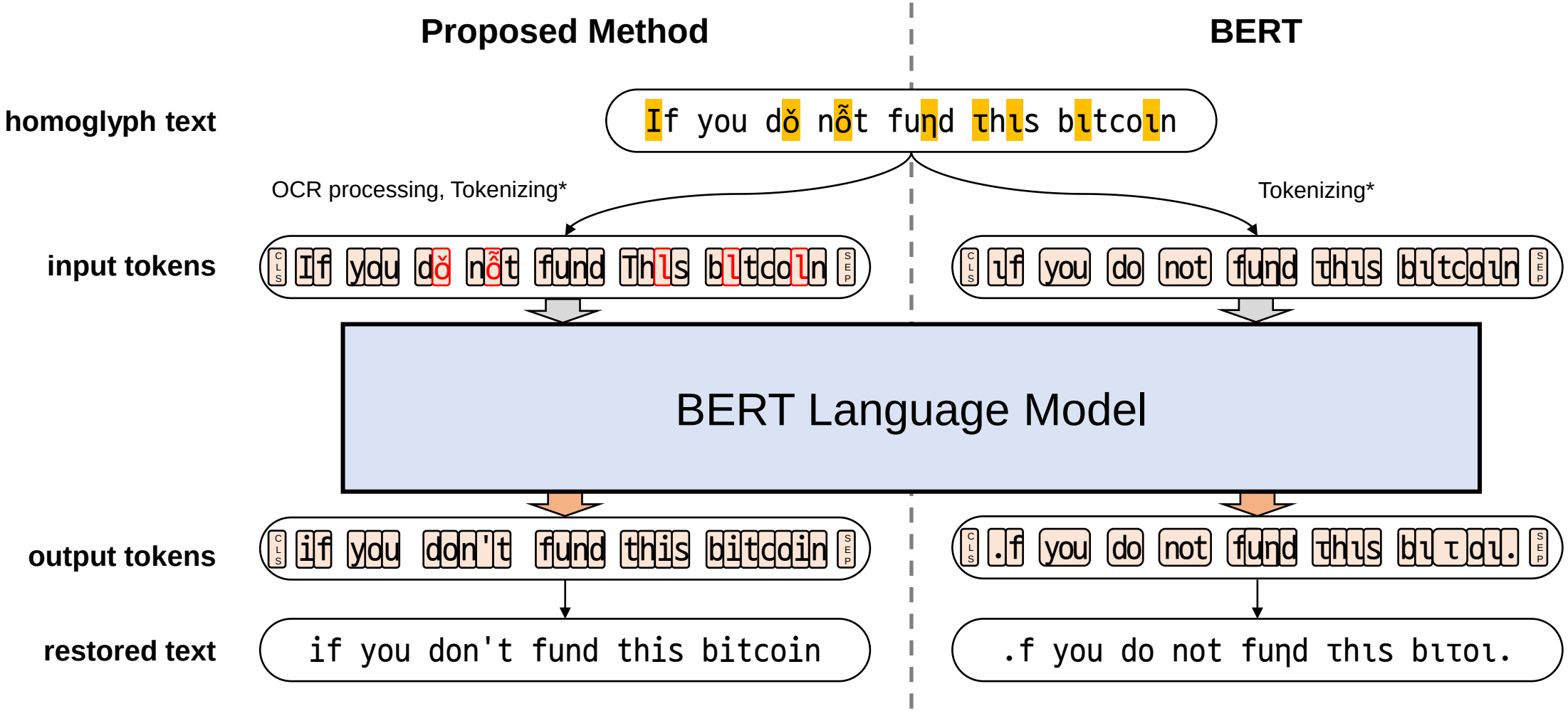
Statistic	p value
100.0	2.3643e-05

Finetuned Model Evaluation – Bonferroni–Dunn test (α = 0.05, CD = 0.7899)



Experimental Results

Analysis – A comparison of the proposed method and BERT



* Tokenizing has a Unicode normalization process.

Experimental Results

Analysis – qualitative analysis

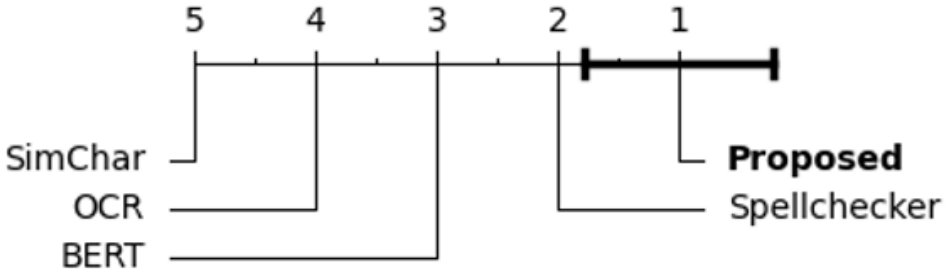
Original Text	Ground Truth	Proposed Method	BERT model
ī am well aware is yōūr pãss.	i am well aware is your pass.	i am well aware is your pass.	l am well as is your pass.
you have 48 hours to make the p ayment.	you have 48 hours to make the p ayment.	you have 48 hours to make the p ayment.	you have 48 hours to make the p ayment.
l have α unıque pixel wiтhiη тhıs e mαıl message	i have a unique pixel within this e mail message	i have a unique pixel within this e mail message	i have a unlqueid with the thls em all message
all íпgeиious ỉs sìмple.	all ingenious is simple.	all ingenious is simple.	all ingenlous ls simple.
you'll mακe тhe ραymeηt via βitco ıη to тhe below αddress	you'll make the payment via bitcoi n to the below address	you'll make the payment via bitcoi n to the below address	you. ll make the paymeat via bitc an to the above address

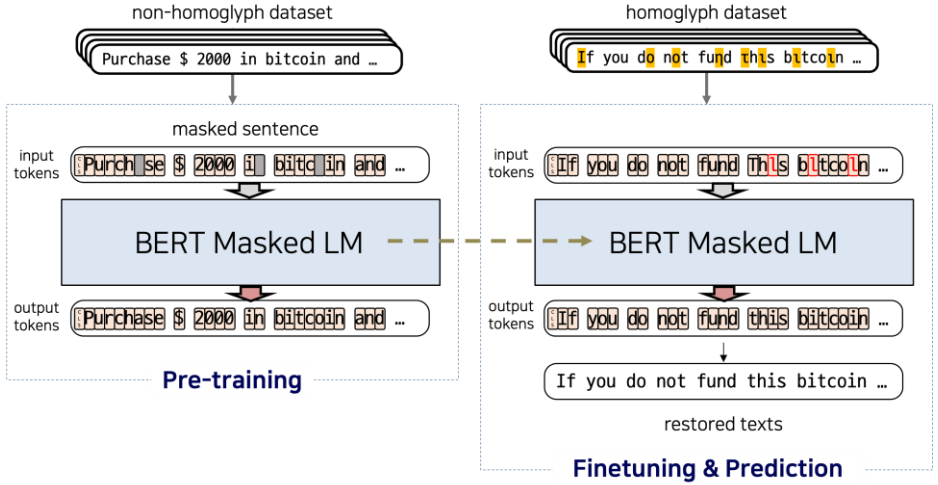
- homoglyph
- wrong word

- Since the BERT model Unicode normalization is being processed during the tokenization, it is partially effective for homoglyph restoration, but not fo r predefined homoglyphs.
- BERT's sub-word tokenization is basically not suitable for homoglyph restoration because one token contains one or more characters.

Conclusion

Contribution of the Study





1. We achieve higher performance than traditional methods such as OCR, SimChar DataBase, and spell checker.

2. We propose the Character-level BERT algorithm and introduce a dataset for the homoglyph restoration task.

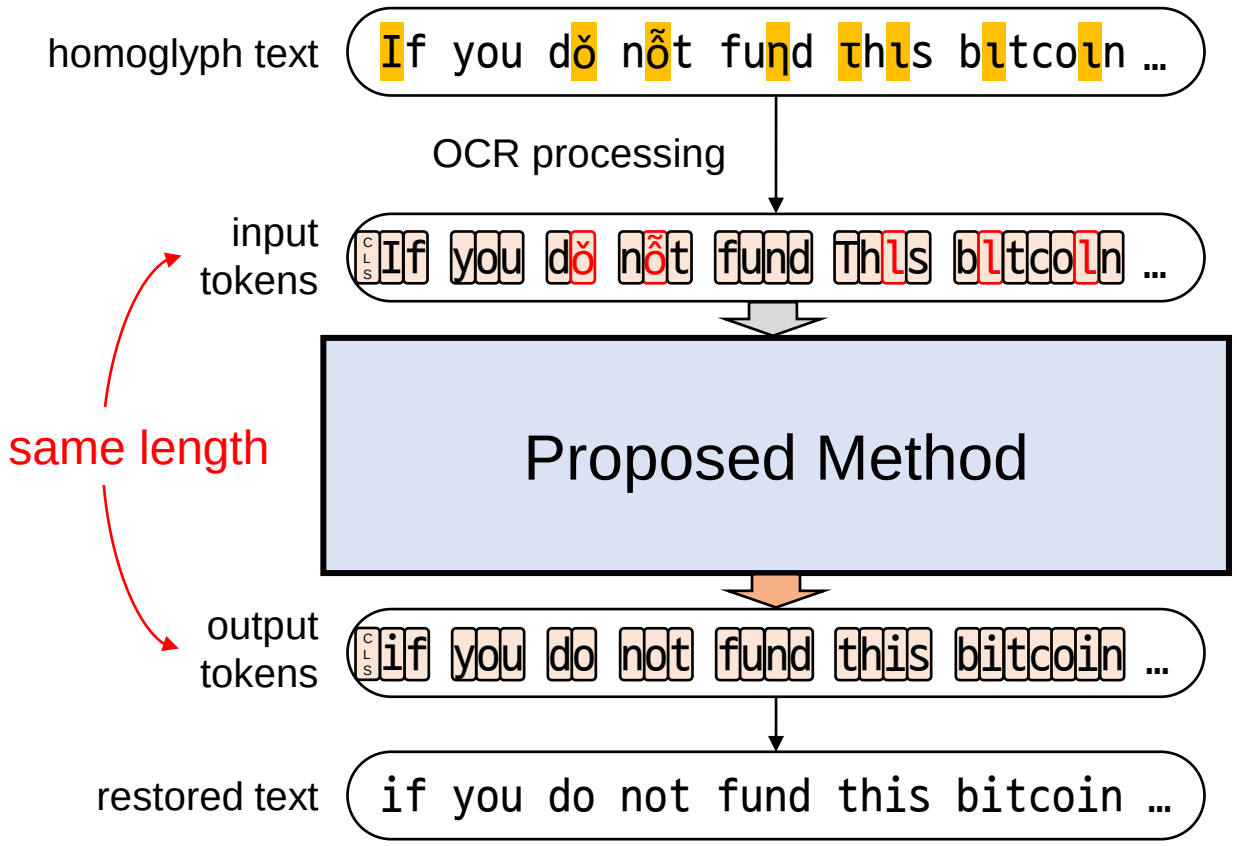
Conclusion

Limitation of Proposed Model

The proposed model is a masked language model, limited in that the input/output sentences should be the **same sequence length**.

- It can be used as a homoglyph of the alpha bet ‘m’ by concatenating the letters ‘r’ and ‘ n’. But the presented model does not solve this case.

The proposed model can be applied as a **sequence-to-sequence** model in future studie s.



References

1. Zhang, W. E., Sheng, Q. Z., Alhazmi, A., & Li, C. (2020). Adversarial attacks on deep-learning models in natural language processing: A survey. ACM Transactions on Intelligent Systems and Technology (TIST), 11(3), 1-41.
2. Boucher, N., Shumailov, I., Anderson, R., & Papernot, N. (2022, May). Bad characters: Imperceptible nlp attacks. In 2022 IEEE Symposium on Security and Privacy (SP) (pp. 1987-2004). IEEE.
3. Unicode Consortium, "Unicode Utilities: Confusables." unicode.org. https://www.unicode.org/Public/security/15.0.0/confusables.txt (accessed 10 September 2022).
4. Suzuki, H., Chiba, D., Yoneya, Y., Mori, T., & Goto, S. (2019, October). ShamFinder: An automated framework for detecting IDN homographs. In Proceedings of the Internet Measurement Conference (pp. 449-462).
5. Sawabe, Y., Chiba, D., Akiyama, M., & Goto, S. (2019). Detection method of homograph internationalized domain names with OCR. Journal of Information Processing, 27, 536-544.
6. Imam, N. H., Vassilakis, V. G., & Kolovos, D. (2022). OCR post-correction for detecting adversarial text images. Journal of Information Security and Applications, 66, 103170.
7. Norvig, P., "How to write a spelling corrector." norvig.com https://norvig.com/spell-correct.html (accessed 10 September 2022).
8. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.
9. Merity, S., Xiong, C., Bradbury, J., & Socher, R. (2016). Pointer sentinel mixture models. arXiv preprint arXiv:1609.07843.
10. Zhu, Y., Kiros, R., Zemel, R., Salakhutdinov, R., Urtasun, R., Torralba, A., & Fidler, S. (2015). Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In Proceedings of the IEEE international conference on computer vision (pp. 19-27).

특허 출원 진행 상황

- P20220246, P20220244, P20220242: 출원 지시 필요
  - 명세서 작성 요청으로 상태 변경 → 오늘 오전 중으로 처리
  - 이후 변리사님이 명세서 초안 업로드
  - 그 이후 출원 지시로 상태 변경 요청 필요

- P20220254
  - 저번 주 금요일 변리사님과 명세서 초안 작성을 위한 미팅 진행함
  - 변리사님이 명세서 초안 작성 중
  - 책임자 이재성 교수님으로 변경 → 오늘 오전 중으로 처리

To-Do List

- 논문작성 계속 진행
- elsevier 양식 재확인
- PPT 내용 수정
  - ✓ Friedman test의 critical value 넣기
  - ✓ Contribution에 in depth analysis 추가

Thank you

Appendix

A. Proposed Model Information

Character-level BERT

- Number of parameters: 86,720,369
- Max length of input tokens: 512
- Vocab size: 881

Appendix

B. Pre-trained Model Evaluation

Evaluated with the **whole test sentences**

Character level BERT is only pretrained with the **WikiText-103, BookCorpus & spam mail datasets**.

Algorithm	Word Accuracy
Proposed (not finetuned)	**0.9516 ± 0.1289**
Normal BERT	0.6713 ± 0.2679 ▼
OCR	0.6519 ± 0.2976 ▼
SimChar DB	0.5506 ± 0.3627 ▼
Spell Checker	0.9090 ± 0.1620 ▼

Word level accuracy and T-test with the accuracy of the proposed method

▼ means that the algorithm has a significantly lower performance than the proposed algorithm based on the t-test.

▼: $p < 0.01$

Appendix

C. Test Output Examples of Pretrained Model

Example 1

- input sentence L Neeԁ y0UR TOTɑı ɑTTeNTıOɴ fOR The c0MıNg TweNTY-fOUR hRS,
- output sentence : i need your total attention for the coming twenty - four hrs,

Example 2

- input sentence : ɑftêr thät, i hȧvë ṡtȧrtèd trȧcкiṅġ ẏȯur ịṅtérпët ȧctívítìêṡ.
- output sentence : after that, i have started tracking yo ur internet activities.

Appendix

C. Test Output Examples of Pretrained Model

Example 3

- input sentence : do ṅȯt trɣ tȯ coṅtáct the ṗolḭćê àɴd ȯtɳêr sëcurity servḭces.
- output sentence : don't try to contact the police and other security services.

Example 4

- input sentence : yøuř åccøuňt wås åttåckěd.
- output sentence : yogr account was attacked.

Data Imputation for missing value

Hyejin Baek

bhj4208@gmail.com

19th Oct, 2022

Contents

1. To Do List from Previous Seminar
2. Experiment
3. NLP(unknown token)
4. To Do List

To Do List from Previous Seminar

- ✓ Dataset 20-25개 이상 사용하여 실험 진행

  → Dataset 1개 추가하여 실험 진행

- ✓ TabNet, NLP(unknown token) 관련 search

  → TabNet 진행 중

- ✓ KNN imputation method 이용하여 실험 진행

  → 진행 완료

- ✓ Cross validation 진행

  → 진행 완료

Experiment

XGBoost

1. Breast_cancer

	20%	40%	60%	80%
Zero	0.975 ± 0.001	0.982 ± 0.003	0.989 ± 0.002	0.992 ± 0.007
KNN	0.967 ± 0.010	0.956 ± 0.019	0.955 ± 0.003	0.959 ± 0.007

2. Abalone

	20%	40%	60%	80%
Zero	0.413 ± 0.004	0.549 ± 0.004	0.704 ± 0.002	0.852 ± 0.004
KNN	0.985 ± 0.000	0.531 ± 0.003	0.694 ± 0.006	0.849 ± 0.004

NLP(unknown token)

NLP(unk) & missing value

- Subword tokenizer

  → unknown token의 수를 줄이기 위해서 사용

  1. Byte Pair Encoding(BPE) (Neural Machine Translation of Rare Words with Subword Units(arXiv, 2016)
  2. SentencePiece (SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing (arXiv, 2018)
  3. SubwordTextEncoder
  4. HuggingFace Tokenizer

- Missing value imputation

  1. KNN
  2. Zero/Mean and so on

To Do List

- ✓ TabNet 분석 후 실험 진행
- ✓ Experiment 코드 보완 및 연구 진행
- ✓ Nearest Neighbors/Machine Learning 연구 진행

Thank you

Gastrointestinal track Polyp Segmentation

Goeun Kim

goeun678@gmail.com

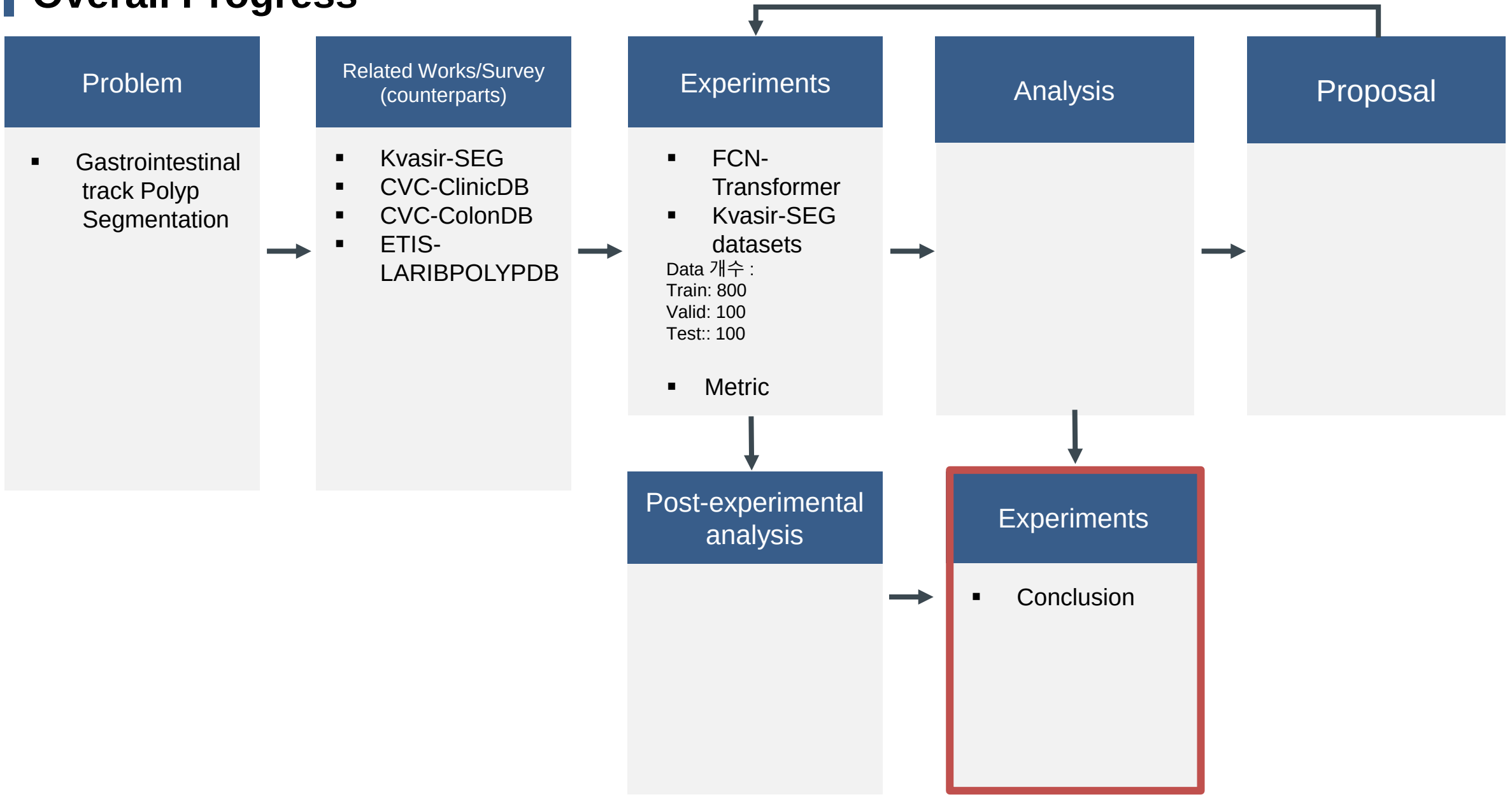
19th Oct, 2022

Contents

To Do List from Previous Seminar

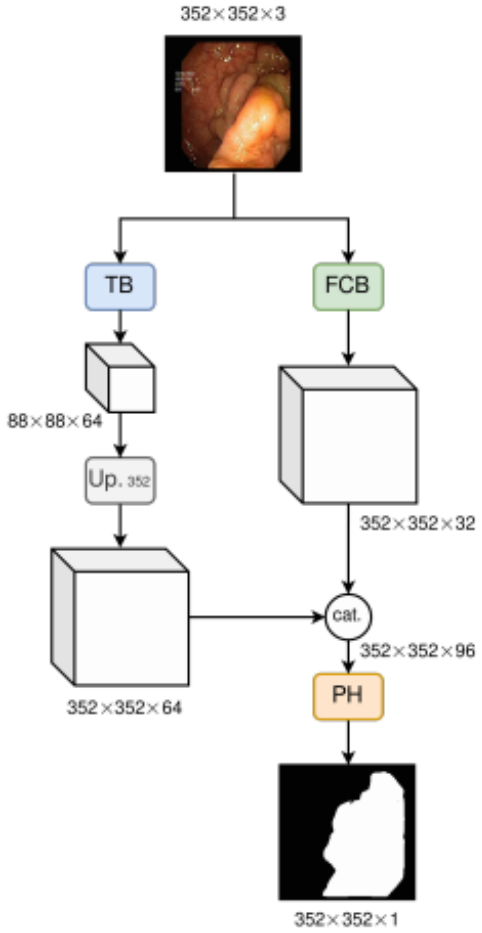
- ✓ To do list 1
  - ➢ GI tract polyp segmentation Public dataset 조사
- ✓ To do list 2
  - ➢ Medical Image polyp segmentation Deep Learning model 조사 , Paper review
- ✓ To do list 3
  - ➢ GI Tract Segmentation 학습 진행

Overall Progress

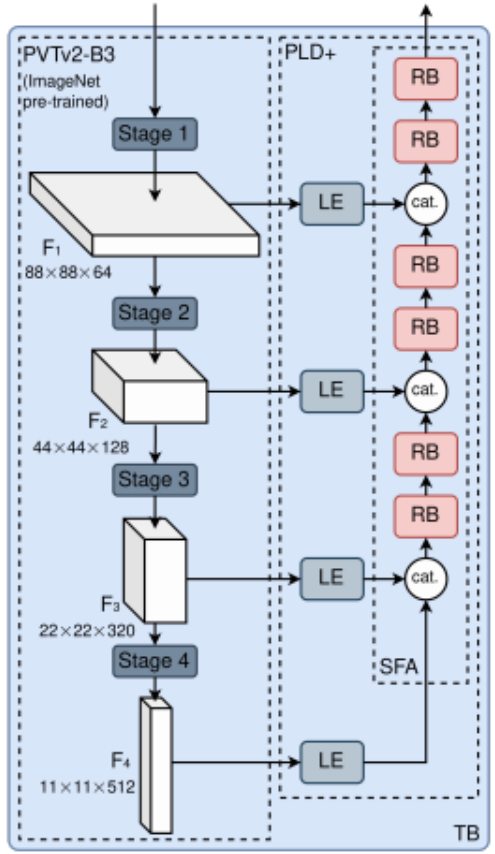


Introduction

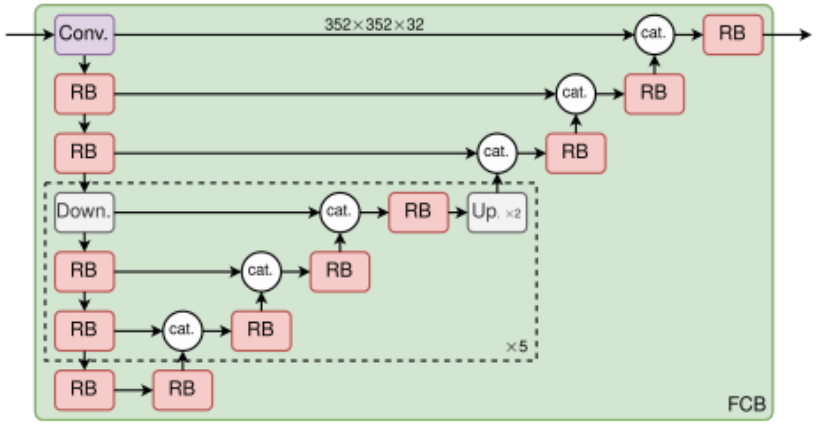
Proposed model : **FCN-Transformer Feature for Polyp Segmentation**



The architectures of FCBFormer



TB (Transformer Branch)



FCB (Fully convolutional Branch)

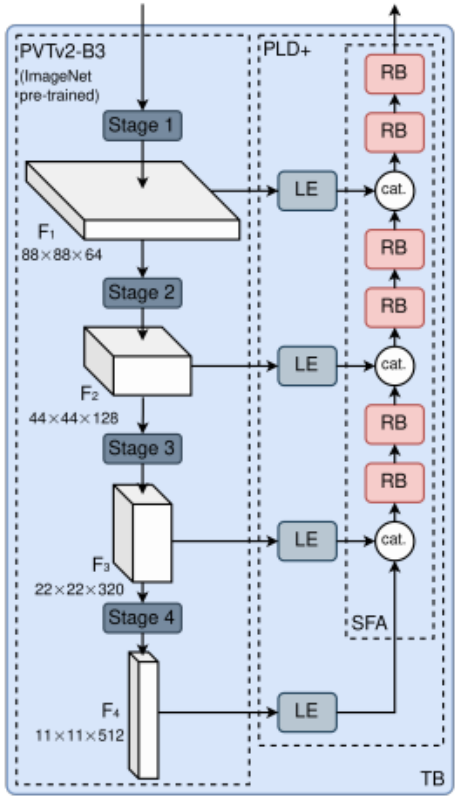
The architecture, named the **Fully Convolutional BranchTransFormer (FCBFormer)** uses two parallel branches which both start from a "h × w" input image:

- **Encoder**: **transformer branch** (**TB**) which returns reduced-size (h/4 × w/4) feature maps.(use the PVTv2 [34] for the image encoder)
- **Decoder**: **a fully convolutional branch (FCB)** which returns full-size (h×w) feature maps.

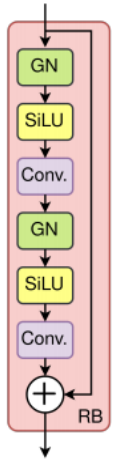
FCN-Transformer Feature Fusion for Polyp Segmentation (Image and Video Processing (eess.IV), Computer Vision and Pattern Recognition (cs.CV), Machine Learning(cs.LG)2022)

Introduction

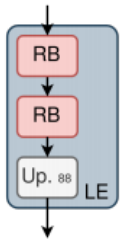
Proposed model : **FCN-Transformer Feature for Polyp Segmentation**



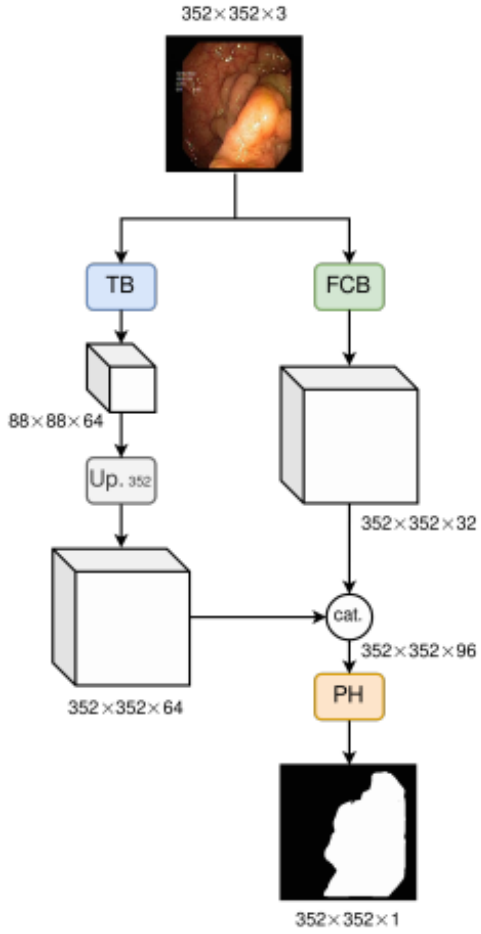
TB (Transformer Branch)



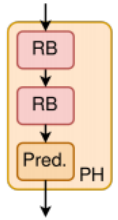
RB (Residual Block)



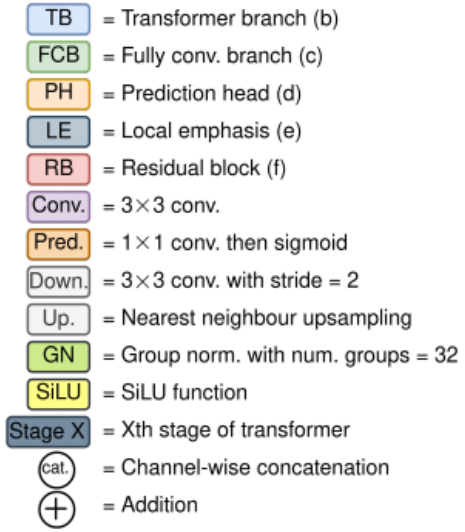
LE (Improved Local Emphasis)



The architectures of FCBFormer



PH (Prediction Head)



FCN-Transformer Feature Fusion for Polyp Segmentation (Image and Video Processing (eess.IV), Computer Vision and Pattern Recognition (cs.CV), Machine Learning(cs.LG)2022)

Related Work

Experimental Results

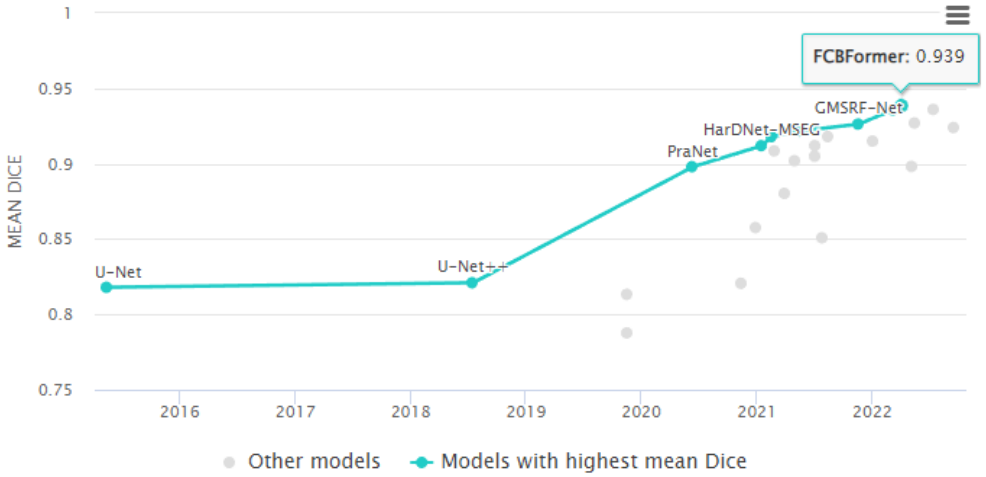


Figure 1. Experimental Results for Kvasir-SEG Dataset

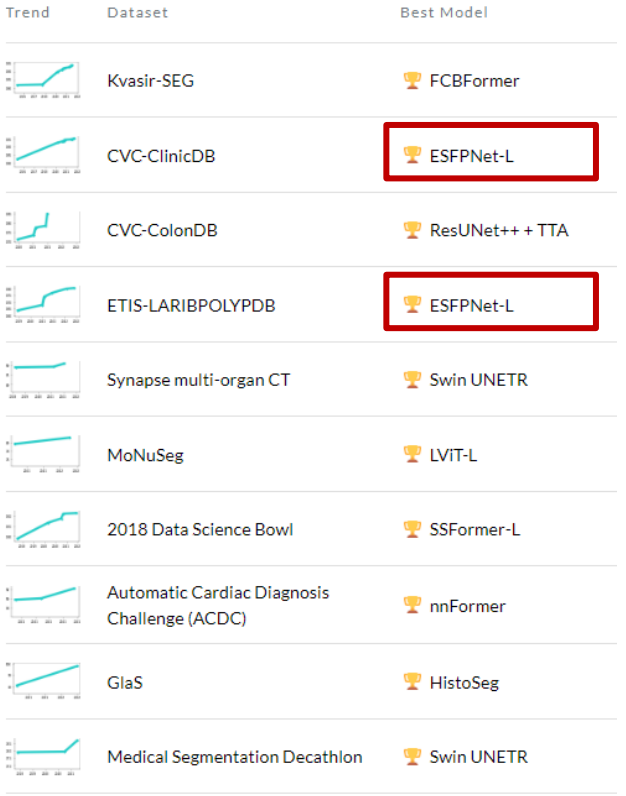


Figure 2. Best Model for Other Medical Image Segmentation dataset

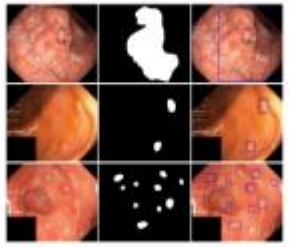
Table 1. Experimental Results for Kvasir-SEG Dataset

Algorithms	mean Dice	mIoU
FCBFormer	**0.9385**	**0.8903**
ESFPNet-L	0.936	0.891
SSFormer-L	0.9357	0.8905
ColonFormer	0.927	0.877
GMSRF-Net	0.9263	0.8843

Related Work

Medical Image Segmentation Dataset

Kvasir-SEG

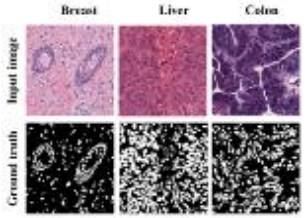


open-access dataset of **gastrointestinal polyp images** and corresponding segmentation masks, The Kvasir-SEG dataset (size 46.2 MB) contains **1000 polyp images** and their corresponding ground truth from the Kvasir Dataset v2. The resolution of the images contained in Kvasir-SEG varies from **332x487 to 1920x1072 pixels.**

CVC-ClinicDB

open-access dataset of **612 images** with a **resolution of 384×288 from 31 colonoscopy sequences**. It is used for medical image segmentation, in particular polyp detection in colonoscopy videos.

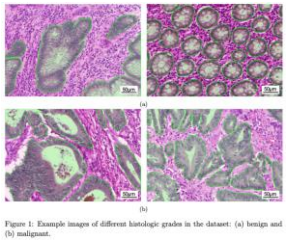
MoNuSeg



The dataset for this challenge was obtained by carefully annotating tissue images of several patients with tumors of different organs and who were diagnosed at multiple hospitals.

- training data : **30 images and around 22,000 nuclear boundary annotations.**
- test data: additional **7,000 nuclear boundary annotations.**

GlaS

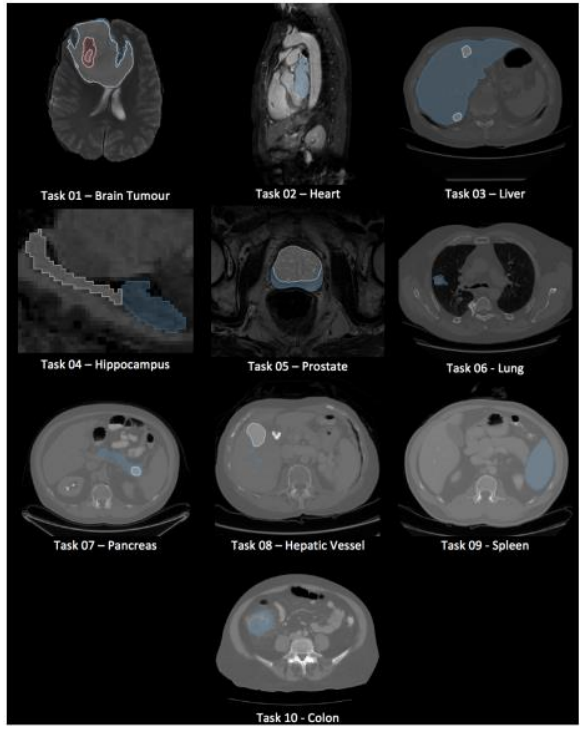


GlaS means **Gland Segmentation in Colon Histology Images Challenge.**
The dataset used in this challenge consists of **165 images** derived from 16 H&E stained histological sections of stage T3 or T42 colorectal adenocarcinoma

Related Work

Medical Image Segmentation Dataset

Medical Segmentation Decathlon



collection of medical image segmentation datasets.
It contains a total of **2,633 three-dimensional images** collected across multiple **anatomies of interest**, multiple modalities and multiple sources.

Related Work

Medical Image Segmentation Model Survey

	저자	연도	저널	제목	요약	별칭
1	Edward Sa	2022.08	Recognitio n (cs.CV), Machine Learning (cs.LG)	Feature Fusion for Polyp Segmentat ion	we propose a new architecture for full-size segmentation which leverages the strengths of a transformer in extracting the most important features for segmentation in a primary branch, while compensating for its limitations in full-size prediction with a	FCBFormer
	Qi Chang 오	2022.07	Image and	autofluore scence bronchosc opic video	We propose a framework for synchronizing multiple bronchoscopic videos to enable interactive multimodal analysis of bronchial lesions. Our methods first register the video streams to a reference 3D chest computed-tomography (CT) scan to produce	ESFPNet-L
3		2022.06	Pattern Recognitio n (cs.C	Local Guides Global	we propose a new State-Of-The-Art model for medical image segmentation, the SSFormer, which uses a pyramid Transformer encoder to improve the generalization ability of models. Specifically, our proposed Progressive Locality Decoder can be adap	SSFormer-L
4	Nguyen Th	2022.06	Computer Vision and Pattern Recognitio n (cs.CV)	Method for Colon Polyp Segmentat ion	This paper proposes a novel network, namely ColonFormer, to address these limitations. ColonFormer is an encoder-decoder architecture capable of modeling long-range semantic information at both encoder and decoder branches. The encoder is a li	ColonFormer
5	Abhishek	2021.11	Computer Vision and Pattern Recognitio n (cs.CV)	fusion network for polyp segmentati on	Our proposed network maintains high-resolution representations while performing multi-scale fusion operations for all resolution scales. To further leverage scale information, we design cross multi-scale attention (CMSA) and multi-scale feature selecti	GMSRF-Net

Experimental Settings

- Datasets
  - Kvasir Dataset
  - Number of Samples: 1,000 (train: 800, test: 100, validation:100)

- Model : FCN-Transformer

- Hyper Parameter Setting:

Model	Optimizer	loss	Epochs	Batch size	Learning rate
FCN-Transformer	AdamW	Dice_loss, BCE_loss	100	32	1e-3

To Do List

- ✓ To do list 1
  - ➢ CVC-ClinicDB, CVC-ColonDB, ETIS-LARIBPOLYPDB 등의 위장계 polyp 검출용 데이터를 사용하여 FCN model학습-> 결과 비교

Thank you

Appendix