Revisiting Classification Perspective on Scene Text Recognition

by jo-member

Abstract

현재의 scene text recognition은 seq2seq와 sementation으로 부터 시작되었다.
이전까지는 text단위의 annotation이 필요한 복잡한 적용과 활용으로 이루어져있다.
이 연구는 revisit classification perspective를 사용하여 text recognition은 image classification 문제로 변화시켰다.

Classification의 pipeline은 매우 단순하고 직관적이기 때문에 word level의 annotation만을 필요로 한다. 우리는 CSTR을 사용했다. CSTR model은 CPNet(Classification perspective network)과 SPPN(seperated conv with global average pooling prediction network)로 이루어져있다. CSTR은 classification model만큼 단순한 구조를 띄고있어 적용과 활용이 매우 쉽다.

CSTR은 많은 부문에서 sota를 달성하고있다. code는 https://github.com/Media-Smart/vedastr 에 있다.

1. Introduction

많은 용도로 쓰이는 STR
STR은 2개의 categoty로 divide할 수 있다.

1. Segementation based method
2. Seq2seq based method

Seq2seq based model또 크게 CTC based method과 attention bases method, transformer-based method로 나누어진다

Segmentation based methods는 2가지 step으로 또 나누어진다

1. character segmentation
2. character recognition

Seq2seq based model은 전체 text line을 한번에 인식하고 각각에 attention을 사용하여 character segementation은 피하고 있다.

1. Image encoding : input image를 CNN, RNN, transformer encoder또는 이들의 조합으로 image를 feature의 sequence로 encoding한다
2. Sequence decoding : 이러한 encoding된 vector들을 CTC나 attention based RNN이나 transformer decoder로 decoding한다.

Segmentation based model은 간단하나, character level의 annotation이 필요하며 길고 복잡한 pipeline을 가지고 있다.

이들말고도 STR 문제를 classification 문제로 정의한 CHAR이라는 논문에서는 k갯수의 character를 max두고 classiciation을 하고있다. CHAR의 model은 4개의 CNN layer와 2개의 FCN으로 이루어져있다. (FCN은 k개의 multi class classification이다)

이러한 CHAR model은 image classification model과 같이 train시키기가 매우 쉽다. 그러나 정확도는 매우 낮다.

Classification based method는 간단하기 때문에 매우 매력적이다.
이러한 Classification based method의 낮은 정확도를 해결한 CSTR model을 제안한다.

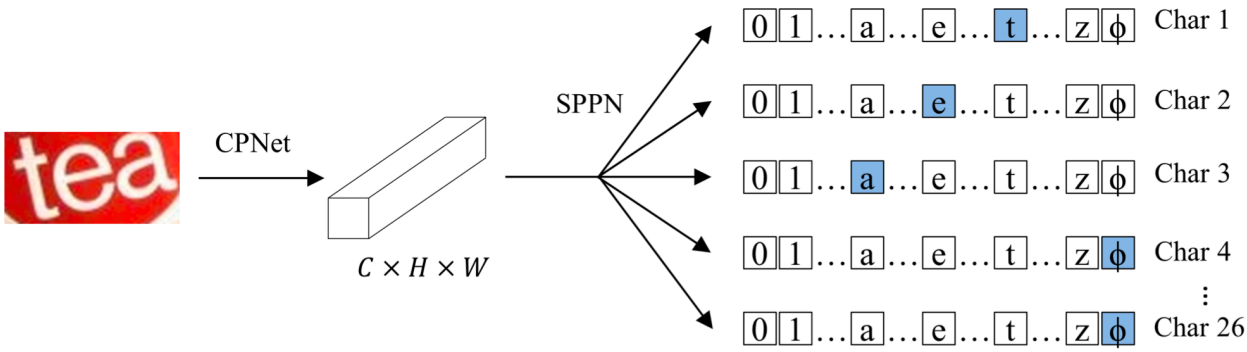
우리는 prediction network로 SPPN이라는 network를 고안해내었다. SPPN은 character의 위치를 encode하기 위해 sementic information을 포함하고 있다.

i개의 classification head가 i개의 image word seqence를 예측하는 이전의 논문들의 방법론을 담고있다. SPPN은 low back propagating computation을 활용하여 CTC보다 잘 작동하게한다.

그리고 backbone network로 CP-Net이라는 새로운 model을 design했다.

- Use CE as loss function

2. Classification Perspective on STR



3. CSTR

구조는 매우 간단하다. Backbone과 prediction network

3.1. Backbone Network

기본적인 backbone model로는 ResNet을 사용하고 있다. input과 output의 차원이 서로 다를때 각각의 잔차 block마다, 1x1 conv를 이용한 projection shrotcut을 사용한다. 이러한 ResNet의 구조를 약가 바꾼것이 CPNet이다. CPNet의 전체적인 구조는 아래와 같다.

Table 1. CPNet body architecture. Residual blocks are highlighted with gray background.

Stage name	Type / Stride : Filter Shape
Stage 1	$Conv/s1 : 1 \times 3 \times 3 \times 48$
	$Conv/s1 : 48 \times 3 \times 3 \times 96$
Stage 2	$Pool/s2 : 2 \times 2$
	$\begin{bmatrix} Conv/s1 : 96 \times 3 \times 3 \times 192 \\ Conv/s1 : 192 \times 3 \times 3 \times 192 \end{bmatrix} \times 1$
	$Conv/s1 : 192 \times 3 \times 3 \times 192$
Stage 3	$SADM\text{-}A$
	$\begin{bmatrix} Conv/s1 : 192 \times 3 \times 3 \times 384 \\ Conv/s1 : 384 \times 3 \times 3 \times 384 \end{bmatrix} \times 4$
	$Conv/s1 : 384 \times 3 \times 3 \times 384$
Stage 4	$SADM\text{-}A$
	$\begin{bmatrix} Conv/s1 : 384 \times 3 \times 3 \times 768 \\ Conv/s1 : 768 \times 3 \times 3 \times 768 \end{bmatrix} \times 7$
	$Conv/s1 : 768 \times 3 \times 3 \times 768$
	$\begin{bmatrix} Conv/s1 : 768 \times 3 \times 3 \times 768 \\ Conv/s1 : 768 \times 3 \times 3 \times 768 \end{bmatrix} \times 5$
Stage 5	$SADM\text{-}B$
	$\begin{bmatrix} Conv/s1 : 768 \times 3 \times 3 \times 768 \\ Conv/s1 : 768 \times 3 \times 3 \times 768 \end{bmatrix} \times 3$
	$Conv/s1 : 768 \times 2 \times 2 \times 768$

1. **Depth, Width and Resolution**

stage3와 stage4를 더 깊게 그리고 stage5에 추가적인 잔차 block을 더해주었다. Network의 width관점에서 보면 1.5배 넓혀주었디. model의 용량을 늘려주기위해 image resolution을 32 x 128에서 48 x 192로 늘려주었다.

2. **FPN**

우리의 model에도 stage3,stage4, stage5와 연결된 FPN을 사용하였다.
따라서 우리의 Output feature map의 width와 height은 input image의 1/4이고 channel은 512이다.

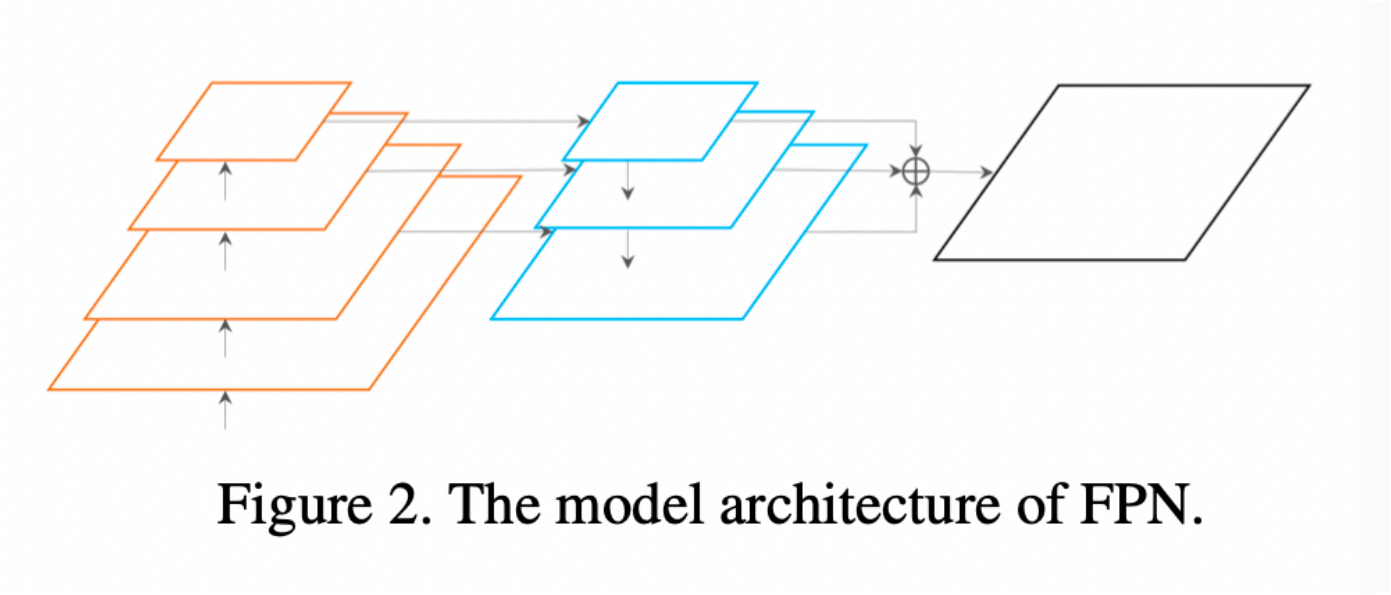


Figure 2. The model architecture of FPN.

3. **CBAM**

Attention은 classification 문제에 매유 효과적이라고 증명이 되어있기때문에 우리도 attention module을 모든 잔차 block에 넣어주었다.

4. **Semantic Aware Downsampling Module**

feature의 downsampling으로 인한 정보의 손실을 줄이기위해 sementic-aware downsampling module을 제안한다. 우리는 이전의 downsampling module인 non local unit을 사용하였다. 아래의 figure은 SADM의 detail을 담고있다.

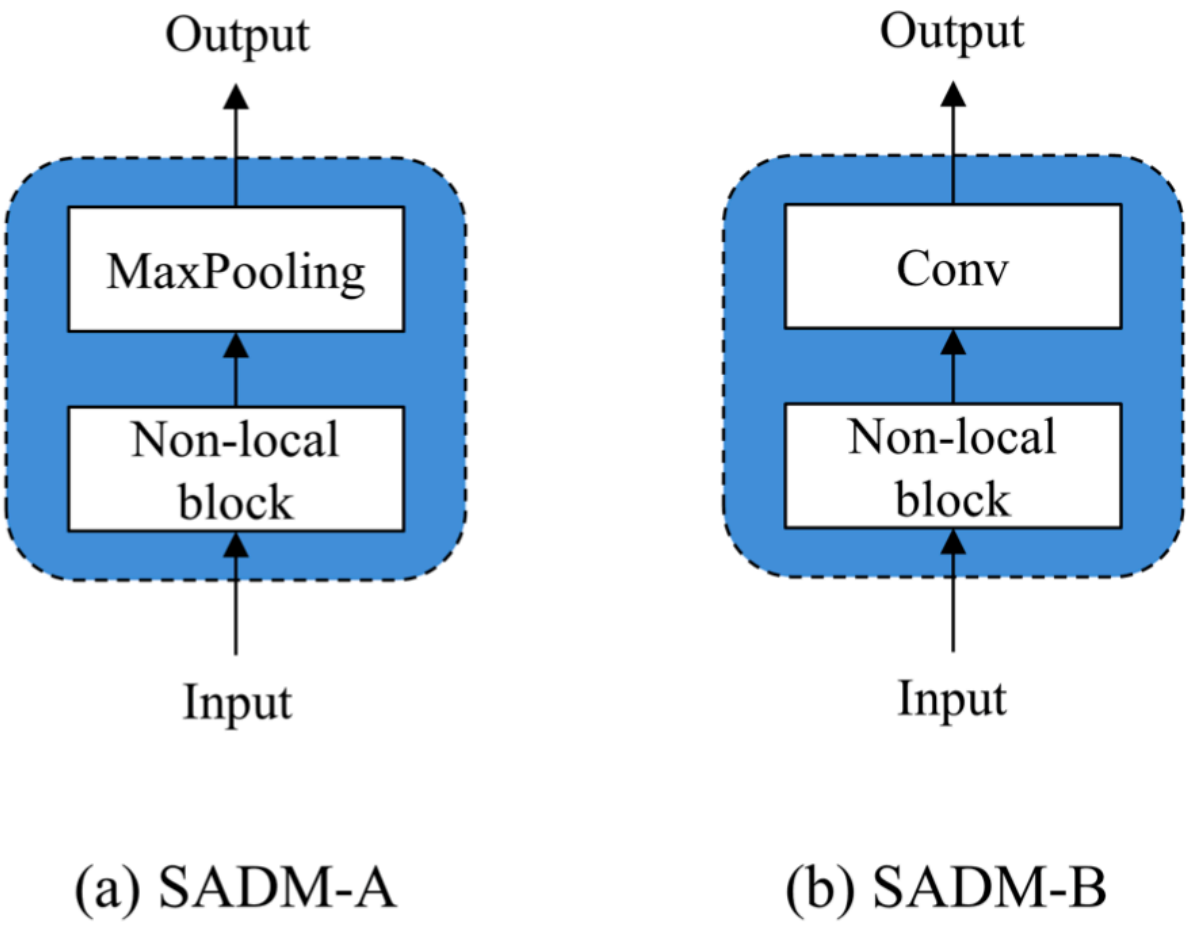
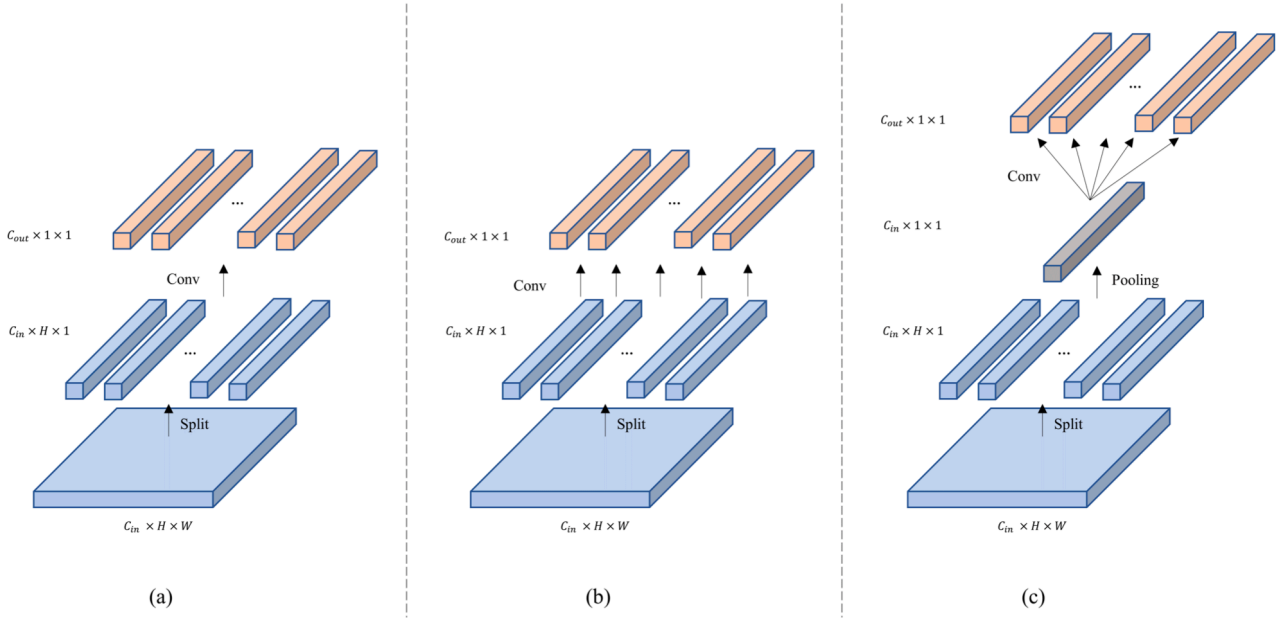


Figure 3. The model architecture of SADM.

3.2. Prediction Network

CSTR의 prediction network는 backbone의 output feature로 부터 textline을 예측하는 역할을 한다. 우리는 3가지 종류의 prediction network를 조사하였다.



- Shared conv prediction network (SHPN)

  CNN layer를 활용한다. Backbone에서 넘어온 feature map의 차원이 C x H x W 라고 할때, W가 이 단어의 최대 길이이다. 이러한 network는 흔히 CTC에서 많이 쓰인다. (a)

- Seperated conv prediction network

  SEPN은 separated CNN layer을 병렬적으로 사용한다. (b)
  C x H x W 에서 CNN layer의 개수는 W개이다. SHPN과 마찬가지로 길이가 W인 단어를 예측한다.

- Seperated conv with global average pooling prediction network (SPPN)

  SPPN은 global average pooling을 사용한다. (c)

  전체적인 이미지의 semantic정보를 넘겨주기 위하여 이러한 구조를 채택함

우리는 이러한 3가지 종류의 prediction network들을 모두 적용해 보았다. 3개를 비교해보면 SHPN과 SEPN은 character를 위치적 정보를 W개의 feature들로 나누어서 담고있다. 우리는 STR을 classification 관점으로 보고있기때문에 각각 나누어진 feature들은 global sementic 정보를 담고있어야한다. 또한 receptive field와 다양한 character 분포가 제한되기 때문에 위의 조건을 만족하지 못한다. 하지만 global average pooling을 이용하면 global sementic information을 모두 잘 담을수 있다. CHAR은 FCN을 이용한다(instead of CHAR). FCN은 아주 많은 parameter를 가지기 때문에 overfitting의 위험이 있다. CHAR와 비교했을때 2개의 SPPN의 장점은 global한 정보를 담고있는것과, 많은 parameter를 사용하지 않는다는것이다. 따라서 우리는 SPPN을 사용하고 있다.

4. Implementation detail

- General Setting
  - Size of image = 48 x 192
  - Number of character classas = 37 = 26 + 10 + 1
  - Max length = 25

- Data Augmentation
  - motion blur
  - gaussian noise

    - color jitter

- Model training
    - label smoothing
    - warm up
    - batch size = 192
    - Adadelta optimizer
    - initial lr = 1
    - The model is totally trained for 420k iterations and the learning rate is decreased 10−1 and 10−2 at 150k iterations and 250k iterations.
- Model Testing
    - No TTA like beam search
    - Greedy 알고리즘
-