Seq2Seq 모델을 이용한 이미지 시퀀스 단어 추론

서정민
서울대학교
대한민국
jmseo1204@snu.ac.kr

Abstract

Abstract

시퀀스 예측은 다양한 산업 분야에서 중요한 문제로, 이를 해결하기 위해 CNN, RNN, Seq2Seq 모델을 사용한 파이썬 모듈을 개발했습니다. 이 연구의 목적은 이러한 모델들이 각각 데이터의 공간적 및 시간적 패턴을 어떻게 학습하는지 조사하고, 특히 Seq2Seq 모델의 성능을 향상시키는 것입니다. 이를 위해, 각 모델의 구조를 설계하고 검증 및 챌린지 데이터셋에서 테스트한 결과, 높은 성능을 보였습니다. 이러한 결과는 개발된 모델이 높은 일반화 성능을 가지며, 산업 현장에서의 예측 유지보수, 시계열 예측, 자연어 처리 등 다양한 문제에 효과적으로 적용될 수 있음을 시사합니다.

1. Introduction

시퀀스 예측 문제는 금융, 의료, 제조 등 여러 산업군에서 중요한 과제로 자리잡고 있습니다. 기존의 CNN, RNN, Seq2Seq 모델은 이러한 문제를 해결하는 데 널리 사용되고 있으며, 각 모델은 데이터의 다양한 특성을 학습하는데 강점을 가집니다. 그러나 Seq2Seq 모델의 경우, 인코더의 마지막 히든 벡터 하나만 디코더로 전달하는 방식에서 발생하는 정보 손실 문제가 있으며, RNN 계열 모델은 시퀀스 길이가 길어질수록 정보를 잊어버리는 문제가 존재합니다.

본 연구는 이러한 문제를 해결하기 위해 RNN 기반의 Seq2Seq 모델의 일반화 성능을 개선하는 것을 목표로 합니다. 이를 위해 인코더의 출력 벡터를 디코더의 입력에 포함시키는 방식을 제안합니다. 이 접근 방식은 정보 손실을 막을 뿐만 아니라 RNN 계열 모델이 시간이 지나면서 정보를 잊어버리는 문제도 해결할 수 있을 것으로 예상됩니다. 이러한 가설을 바탕으로, 다양한 실험을 통해 제안된 방법의 효과를 검증하고자 합니다.

.

2. Method

2.1. Normal Task

일반 과제를 위해 Seq2Seq 모델을 사용하였으며, 다음과 같은 주요 기능과 구조적 선택을 통해 성능 최적화를 목표로 하였습니다.

2.1.1. 입력 시퀀스 이미지의 채널 축소

데이터를 분석한 결과, 이미지에 색이 포함되어 있지만, 이 문제는 이미지 색과 상관없이 이미지 시퀀스가 어떤 단어인지 맞추는 것입니다. 따라서

3채널 입력 이미지를 1채널로 합산하고 minmax 스케일링을 적용하여 28x28x1 크기의 데이터로 변환하였습니다.

2.1.2. 양방향 LSTM 사용

RNN 계열 모델 중 장기 기억을 보존하는 LSTM을 사용하였고, 입력된 글자 시퀀스의 순서 이상을 탐지하기 위해 현재 시점 이전과 이후 시퀀스 정보가 모두 필요하므로 양방향 LSTM을 사용했습니다. 여기서 인코더의 마지막 히든 벡터 레이어는 양방향이기에 2배가 되는데, i번째 레이어의 정방향 히든 벡터와 역방향 히든 벡터를 연결하여 해당 레이어의 모든 정보를 담도록 하였습니다. 실제로 연결하지 않고 디코더의 레이어 수를 인코더의 2배로 설정했을 때보다, 연결하여 디코더의 입력 차원을 인코더의 히든 차원의 2배로 만드는 대신 레이어 수를 유지한 모델의 성능이 더 좋았습니다.

2.1.3. 간단한 CNN 모델 사용

이 문제의 입력 이미지는 28x28x3의 작은 데이터로, 복잡한 물체가 아닌 단순한 알파벳을 포함하고 있습니다. 따라서 깊은 CNN 모델을 사용하면 과적합이 발생할 수 있으므로, 학습 효율성을 위해 3개의 CNN 레이어와 3개의 Affine 레이어를 이용한 간단한 CNN 모델을 만들었습니다.

2.2. Challenge Task

처음에는 기존 Seq2Seq 모델에서 decoder에서 output vector을 계산할 때, decoder hidden_state의 encoder의 output과의 alignment를 Attention으로 계산하여 decoder input에 concat해주는 구조의 Seq2seq attention 모델과 Transformer 기반 encoder, decoder 모델을 각각 사용했지만 학습 진행이 잘 안되는 문제가 있었습니다. 예상되는 원인은 입력 이미지가 너무 단순하고 출력도 단순한 문자 변환이었기 때문에 과적합 문제가 발생한다는 것이었습니다. 하지만 그렇다고 인코더의 RNN 셀 하나하나의 정보를 context 벡터 하나로 압축하는 것은 정보 손실이 너무 크다고 생각했습니다. 따라서 Seq2Seq Attention 모델에서 착안하여, 인코더의 각 셀의 출력을 활용하는 아이디어를 떠올렸습니다. 본 문제는 입력의 순서를 맞추는 것이기 때문에 encoder의 어떤 cell이 decoder cell에 align되는지를 Attention으로 추론할 필요 없이 encoder의 i번째 셀이 decoder의 i번째 셀에 의미적으로 대응된다는 점에서 encoder의 i번째 출력 벡터를 디코더의 i번째 셀의 입력에 concat하여 사용하는 전략을 세웠습니다.

2.3. 하이퍼파라미터

학습률 (lr)은 여러 번의 실험 결과 0.003~0.004 부근에서 최적의 학습이 이루어졌습니다.

히든 차원 (hidden_dim)은 256보다 작으면 과소적합, 크면 과적합이 발생하는 것을 확인하였습니다.

RNN 레이어 수 (n_rnn_layers)은 2 이상의 값을 설정하면 과적합으로 인해 손실 감소율이 상당히 느려지고 손실이 줄어들지 않는 것을 실험으로 확인하였습니다.

RNN 드롭아웃 (rnn_dropout)은 0.5 근방에서 최적의 정확도를 보였으며, 더 낮으면 학습 속도는

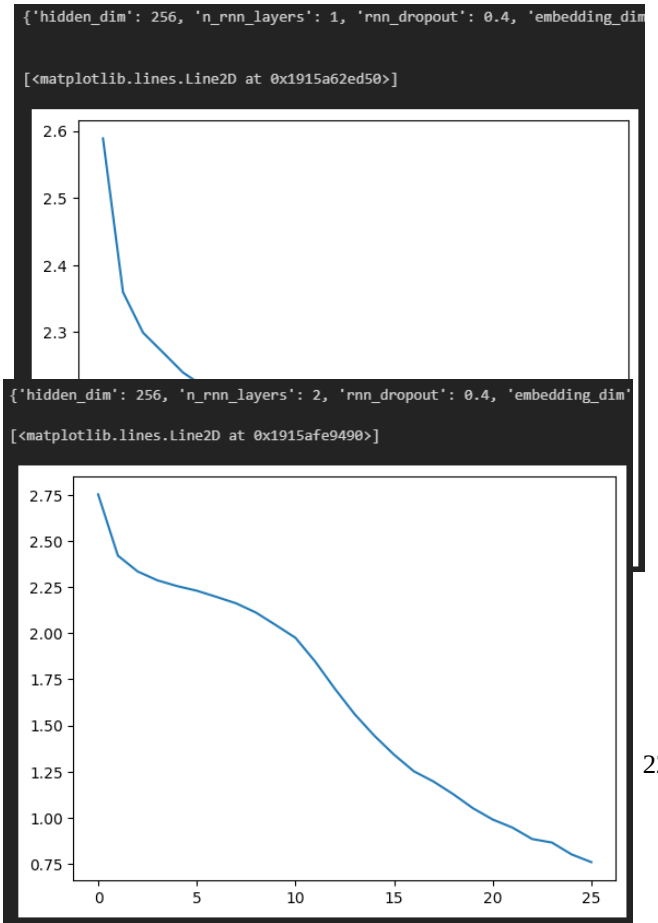
빨라지지만 검증 정확도가 낮아지면서 드롭아웃이 과적합을 막아주는 것을 확인할 수 있었습니다.

임베딩 차원 (embedding_dim)은 90으로 설정하였으며, 64~128 사이가 아니면 손실 감소율이 느려지는 것을 확인하였습니다.

3. Experiments

3.1. 하이퍼파리미터 실험

모델에 가장 적합한 하이퍼파라미터를 설정하기 위해 hidden_dim, num_layers, dropout_p, embedding_dim 총 4개의 하이퍼파라미터의 최적값을 탐색했습니다. 파라미터 탐색은 나머지 모든 하이퍼파라미터를 고정시켜두고, 특정 파라미터를 조작하면서 진행했습니다. 엄밀하게는 acccuracy로 비교해야하지만 제한된 시간 내에 학습을 해야하기에 특정 에폭동안 loss가 감소하는 속도를 비교하였습니다. 다음은 그 사례입니다.



위 plot은 num_layers=1로 했을 때 1 epoch동안 loss의 감소를 나타냈고, 아래 plot은 num_layers = 2로 설정했을 때 1 epoch동안 loss의 감소를 나타냅니다. 이때 num_layers =1에서 1 epoch을 거쳤을 때 loss는 2 이상인 반면, num_layers=2에서는 loss가 1 미만이므로 num_layers=2가 더 좋은 하이퍼파라미터임을 확인할 수 있습니다. 나머지 하이퍼파라미터도 위와 같은 방식으로 여러 번의 실험을 통해 가능한 한 최적의 수치를 구했습니다.

그 결과, Normal task를 위한 Seq2SeqModel의 최적 하이퍼파라미터는 다음과 같습니다.

```
'hidden_dim': 256,
   'n_rnn_layers': 1,
   'rnn_dropout': 0.5,
   'embedding_dim': 90
```

그 결과 얻은 수치는 다음과 같습니다.
Normal task accuracy: 68
Challenge task accuracy: 70

4. 결론

본 연구에서는 이미지 시퀀스 데이터를 처리하여 해당 시퀀스가 표현하는 단어를 예측하는 문제를 다루었습니다. 이를 위해 Normal task와 Challenge task로 나누어 각기 다른 모델 구조와 학습 방식을 제안하고 최적화하였습니다.

Normal task에서는 입력 이미지의 채널을 축소하여 데이터 전처리를 수행하고, 양방향 LSTM과 간단한 CNN 모델을 결합하여 높은 성능을 보이는 모델을 설계했습니다. 이 과정에서 인코더의 마지막 hidden

벡터를 concatenate하여 디코더의 성능을 향상시키는 전략이 유효했음을 확인했습니다.

Challenge task에서는 Seq2Seq Attention 모델과 Transformer 모델을 초기 시도하였으나, 과적합 문제로 인해 성능이 저하되는 현상을 발견했습니다. 이를 극복하기 위해 인코더의 각 셀이 방출하는 정보를 디코더의 입력으로 직접 사용하는 방식을 제안하였고, 이 접근법이 정보 손실을 줄이고 성능을 개선하는 데 효과적이었습니다.

모델의 최적화를 위해 다양한 하이퍼파라미터 설정을 실험한 결과, 적절한 learning rate, hidden_dim, n_rnn_layers, rnn_dropout, embedding_dim 값을 찾아내어 학습 효율성과 정확도를 동시에 달성할 수 있었습니다.

본 연구의 결과로 제안된 모델들은 Valid와 Challenge Valid 데이터셋에서 높은 성능을 보였으며, 이는 본 문제와 관련된 다른 유사한 문제에도 적용 가능할 것으로 기대됩니다. 특히, 본 연구에서 사용된 데이터 전처리 기법과 모델 최적화 전략은 다양한 산업 분야에서 이미지 시퀀스 데이터를 활용한 예측 모델 개발에 유용하게 적용될 수 있을 것입니다.

앞으로 더 복잡한 이미지 데이터셋이나 다양한 언어 모델과의 결합을 통해 모델의 성능을 더욱 향상시킬 수 있을 것으로 기대되며, 실제 산업 현장에서의 적용 가능성을 높이기 위한 추가 연구가 필요할 것입니다.

Reference