

10장. LSTM 순환 신경망 : 도시 소음 분류 신경망

1. 개요

이 실험에서는 LSTM 순환 신경망을 도시 소음 분류 신경망에 적용해보는 작업을 수행하였다. LSTM은 "Long Short-Term Memory"의 약자로, 순환 신경망(RNN)의 한 종류이다. 시퀀스 데이터와 같이 이전 단계의 정보가 현재 작업에 영향을 줄 수 있는 데이터를 처리하는 데 사용된다. 이 실험에서는 이 실험에서는 LSTM을 활용하여 다양한 윈도우 크기의 객체에 대한 세 가지 모델을 비교하고 분석해보며, 깊은 신경망에서의 소멸 및 폭주 문제에 대해 더 살펴보고자 한다.

2. 실험 결과

2.1 작은 윈도우 사용

(1) 모델 1

구분	내용
조건	- 계층 : RNN계층 - 에포크 횟수 : 100 - interval : 10 - 윈도우 크기 : 10
데이터 정확도	37.4%
학습 시간	410s
결과 화면	Model us_basic_10_10 train started: Epoch 20: cost=1.627, accuracy=0.414/0.260 (78/78 secs) Epoch 40: cost=1.491, accuracy=0.448/0.340 (80/158 secs) Epoch 60: cost=1.429, accuracy=0.465/0.340 (85/243 secs) Epoch 80: cost=1.414, accuracy=0.481/0.270 (84/327 secs) Epoch 100: cost=1.319, accuracy=0.497/0.330 (83/410 secs) Model us_basic_10_10 train ended in 410 secs: Model us_basic_10_10 test report: accuracy = 0.374, (0 secs)

2) 모델 2

구분	내용
조건	- 계층 : lstm 계층(순환벡터 출력) - 에포크 횟수 : 100

	<ul style="list-style-type: none"> - interval : 10 - 윈도우 크기 :10
데이터 정확도	48.6%
학습 시간	1569s
결과 화면	<p>Model us_lstm_10_10 train started:</p> <p>Epoch 20: cost=1.749, accuracy=0.378/0.290 (311/311 secs)</p> <p>Epoch 40: cost=1.561, accuracy=0.448/0.340 (313/624 secs)</p> <p>Epoch 60: cost=1.425, accuracy=0.492/0.430 (318/942 secs)</p> <p>Epoch 80: cost=1.383, accuracy=0.539/0.380 (321/1263 secs)</p> <p>Epoch 100: cost=1.275, accuracy=0.582/0.390 (306/1569 secs)</p> <p>Model us_lstm_10_10 train ended in 1569 secs:</p> <p>Model us_lstm_10_10 test report: accuracy = 0.486, (1 secs)</p>

3) 모델 3

구분	내용
조건	<ul style="list-style-type: none"> - 계층 : lstm 계층(상태벡터 출력) - 에포크 횟수 : 100 - interval : 10 - 윈도우 크기 :10
데이터 정확도	57.7%
학습 시간	1532s
결과 화면	<p>Model us_state_10_10 train started:</p> <p>Epoch 20: cost=1.538, accuracy=0.450/0.460 (300/300 secs)</p> <p>Epoch 40: cost=1.247, accuracy=0.547/0.520 (303/603 secs)</p> <p>Epoch 60: cost=1.119, accuracy=0.600/0.490 (308/911 secs)</p> <p>Epoch 80: cost=0.991, accuracy=0.646/0.430 (310/1221 secs)</p> <p>Epoch 100: cost=0.889, accuracy=0.678/0.530 (311/1532 secs)</p> <p>Model us_state_10_10 train ended in 1532 secs:</p> <p>Model us_state_10_10 test report: accuracy = 0.577, (1 secs)</p>

2.2 큰 윈도우 사용

1) 모델 4

구분	내용
조건	<ul style="list-style-type: none"> - 계층 : RNN계층 - 에포크 횟수 : 100 - interval : 10 - 윈도우 크기 :100
데이터 정확도	59.0%

학습 시간	387s
결과 화면	Model us_basic_10_100 train started: Epoch 20: cost=1.138, accuracy=0.602/0.580 (91/91 secs) Epoch 40: cost=1.015, accuracy=0.649/0.550 (75/166 secs) Epoch 60: cost=0.978, accuracy=0.667/0.590 (74/240 secs) Epoch 80: cost=0.784, accuracy=0.711/0.690 (73/313 secs) Epoch 100: cost=0.845, accuracy=0.700/0.650 (74/387 secs) Model us_basic_10_100 train ended in 387 secs: Model us_basic_10_100 test report: accuracy = 0.590, (0 secs)

2) 모델 2

구분	내용
조건	<ul style="list-style-type: none"> - 계층 : lstm 계층(순환벡터 출력) - 에포크 횟수 : 100 - interval : 10 - 윈도우 크기 : 100
데이터 정확도	72.2%
학습 시간	1463s
결과 화면	Model us_lstm_10_100 train started: Epoch 20: cost=1.229, accuracy=0.593/0.520 (290/290 secs) Epoch 40: cost=0.910, accuracy=0.690/0.650 (291/581 secs) Epoch 60: cost=0.746, accuracy=0.756/0.680 (293/874 secs) Epoch 80: cost=0.704, accuracy=0.772/0.660 (294/1168 secs) Epoch 100: cost=0.618, accuracy=0.806/0.680 (295/1463 secs) Model us_lstm_10_100 train ended in 1463 secs: Model us_lstm_10_100 test report: accuracy = 0.722, (1 secs)

3) 모델 3

구분	내용
조건	<ul style="list-style-type: none"> - 계층 : lstm 계층(상태벡터 출력) - 에포크 횟수 : 100 - interval : 10 - 윈도우 크기 : 100
데이터 정확도	75.8%
학습 시간	1459s

결과 화면	Model us_state_10_100 train started: Epoch 20: cost=0.979, accuracy=0.682/0.640 (288/288 secs) Epoch 40: cost=0.746, accuracy=0.747/0.690 (290/578 secs) Epoch 60: cost=0.609, accuracy=0.794/0.710 (292/870 secs) Epoch 80: cost=0.482, accuracy=0.837/0.740 (294/1164 secs) Epoch 100: cost=0.461, accuracy=0.848/0.730 (295/1459 secs) Model us_state_10_100 train ended in 1459 secs: Model us_state_10_100 test report: accuracy = 0.758, (1 secs)
-------	--

3-1. 추가 분석

(1) 깊은 신경망에서의 소멸 및 폭주 문제

소멸 문제: 역전파 과정에서 계층을 거칠수록 손실 기울기 값이 점점 작아져 초반 계층 학습이 잘 안 되는 현상을 의미한다.

폭주 문제: 계층을 거칠수록 손실 기울기가 점점 커져 초반 계층의 학습이 엉망이 되어버리는 현상을 의미한다.

손실 기울기의 소멸은 학습 진행을 느리게 만드는 정도지만 폭주는 단 한 번만으로도 애써 학습한 파라미터값을 크게 훼손하게 된다.

* 해결 방법

1. 미니배치 데이터를 이용해 문제를 일으킬 소지가 있는 특정 데이터를 다른 데이터와 혼합해 처리하면서 많이 완화할 수 있다.

2. 편미분을 이용한 손실 기울기 계산 과정을 정확히 이해하고 적용하면서 문제 자체가 많이 해소될 수 있다.

요즘은 손실 기울기의 소멸 및 폭주 현상 자체가 그다지 심각한 문제로 거론되지는 않는 상태이며, 신경망이 깊어질수록 좀 더 많은 데이터와 학습 횟수를 필요로 하지만 매우 깊은 신경망 구조도 가능해진다.

3-2. 결론

실험 내용을 표로 정리하면 다음과 같다.

윈도우 크기	계층	정확도(%)	소요시간(s)
10	RNN	37.4	410
	LSTM	48.6	1569
	LSTM+state	57.7	1532

100	RNN	59.0	387
	LSTM	72.2	1463
	LSTM+state	75.8	1459

실험 결과, 윈도우 크기에 따라 정확도와 소요 시간에는 명확한 차이가 나타났다. 윈도우 크기가 증가함에 따라 모델의 정확도가 향상되는 경향이 보였음을 알 수 있다. 작은 윈도우 (크기 10)에서는 RNN이 37.4%의 정확도를 보여주며, LSTM은 48.6%로 약간 더 높았다. 그러나 윈도우 크기가 커짐에 따라 정확도가 크게 향상되었다. 윈도우 크기가 100인 경우, RNN의 정확도는 59.0%로 상승하고, LSTM은 72.2%로 향상되었다. 특히, 윈도우 크기가 커질수록 LSTM에 상태벡터를 추가한 경우 정확도가 가장 높게 나타났다. 윈도우 크기가 100인 경우, LSTM에 상태벡터를 포함한 모델이 75.8%의 더 높은 정확도를 보여주었다.

계층 간에서 윈도우 크기의 영향을 받는 경향을 보였다. 작은 윈도우에서는 RNN과 LSTM 간의 정확도 차이가 크지 않았으나, 윈도우 크기가 커짐에 따라 LSTM이 RNN보다 훨씬 더 높은 정확도를 보여주었다. 또한, LSTM에 상태벡터를 추가한 경우 모든 윈도우 크기에서 가장 높은 정확도를 기록했다. 시간 소요 측면에서는 윈도우 크기가 커짐에 따라 모든 계층에서 학습에 소요되는 시간이 증가했다. 하지만 모든 경우에서 LSTM 계층이 RNN에 비해 학습에 더 많은 시간이 소요되었음을 확인할 수 있다.