

6장. 복합 출력의 처리 방법 : 오피스 31 다차원 분류 신경망

1. 개요


오피스 31 다차원 분류 신경망 실험에서는 도메인 분류와 품목 분류를 동시에 수행하며 둘 이상의 복합 출력을 학습하는 방법에 대해 알아보려고 한다. 다차원 분류는 도메인과 품목을 동시에 판별해주는 이중 선택 분류로, 한 도메인에서 학습시킨 결과를 다른 도메인에 활용하는 기법인 전이 학습과 차이가 있다. 하지만 다차원 분류는 도메인 특성과 품목 특성이 뒤섞여 학습 성과가 좋지 않을 수도 있다는 단점도 존재한다. 이 실험에서는 다층 퍼셉트론 신경망을 사용하며 은닉 계층, 에포크 횟수 조정을 통한 차이를 분석하고, 아담 알고리즘으로 학습을 진행 시 차이 또한 살펴보고자 한다. 이와 더불어 Stochastic Gradient 방법 사용과 아담 알고리즘 사용의 차이 및 전이 학습의 필요성을 분석해보고자 한다.

2 학습 결과

2.1 아담 알고리즘을 사용하지 않은 경우


(1) 모델 1

구분	내용
조건	- 에포크 횟수(epoch_count) : 20 - 학습률(learning rate) : 0.001 - 은닉 계층 개수 = 1
데이터 정확도	도메인 : 66.1%, 상품 : 4.8%
소요 시간	68s

결과 화면	<pre>om1 = AdamModel('office31_model_1', od, [10]) om1.exec_all(epoch_count=20, report=10)</pre> <p>Model office31_model_1 train started: Epoch 10: cost=4.296, accuracy=0.685+0.038/0.720+0.030 (33/33 secs) Epoch 20: cost=4.268, accuracy=0.685+0.037/0.770+0.030 (35/68 secs) Model office31_model_1 train ended in 68 secs: Model office31_model_1 test report: accuracy = 0.661+0.048, (0 secs)</p> <p>Model office31_model_1 Visualization</p>  <p>[도메인 추정결과] 추정확률문포 [65,15,20] => 추정 amazon : 정답 amazon => 0 추정확률문포 [65,15,20] => 추정 amazon : 정답 webcam => X 추정확률문포 [65,15,20] => 추정 amazon : 정답 webcam => X [상품 추정결과] 추정확률문포 [3, 3, 3, 3, 3, 3, 3, 3,...] => 추정 monitor : 정답 ring_binder => X 추정확률문포 [3, 3, 3, 3, 3, 3, 3, 3,...] => 추정 monitor : 정답 phone => X 추정확률문포 [3, 3, 3, 3, 3, 3, 3, 3,...] => 추정 monitor : 정답 ring_binder => X</p>
-------	--


(2) 모델 2

구분	내용
조건	<ul style="list-style-type: none"> - 에포크 횟수(epoch_count) : 50 - 학습률(learning rate) : 0.001 - 은닉 계층 개수 = 1
데이터 정확도	도메인 : 66.1%, 상품 : 4.8%
소요 시간	155s

결과 화면	<pre>om1 = AdamModel('office31_model_1', od, [10]) om1.exec_all(epoch_count=50, report=10)</pre> <p>Model office31_model_1 train started:</p> <p>Epoch 10: cost=4.296, accuracy=0.685+0.035/0.790+0.040 (30/30 secs)</p> <p>Epoch 20: cost=4.268, accuracy=0.685+0.037/0.780+0.020 (32/62 secs)</p> <p>Epoch 30: cost=4.263, accuracy=0.685+0.038/0.810+0.020 (32/94 secs)</p> <p>Epoch 40: cost=4.262, accuracy=0.685+0.037/0.790+0.030 (30/124 secs)</p> <p>Epoch 50: cost=4.260, accuracy=0.685+0.036/0.790+0.050 (31/155 secs)</p> <p>Model office31_model_1 train ended in 155 secs:</p> <p>Model office31_model_1 test report: accuracy = 0.661+0.048, (0 secs)</p> <p>Model office31_model_1 Visualization</p>  <p>[도메인 추정결과]</p> <p>추정 확률분포 [68,13,19] => 추정 amazon : 정답 amazon => 0</p> <p>추정 확률분포 [68,13,19] => 추정 amazon : 정답 amazon => 0</p> <p>추정 확률분포 [68,13,19] => 추정 amazon : 정답 amazon => 0</p> <p>[상품 추정결과]</p> <p>추정 확률분포 [3, 3, 3, 3, 3, 3, 3, 3,...] => 추정 monitor : 정답 stapler => X</p> <p>추정 확률분포 [3, 3, 3, 3, 3, 3, 3, 3,...] => 추정 monitor : 정답 calculator => X</p> <p>추정 확률분포 [3, 3, 3, 3, 3, 3, 3, 3,...] => 추정 monitor : 정답 calculator => X</p>
-------	--


(3) 모델3

구분	내용
조건	<ul style="list-style-type: none"> - 에포크 횟수(epoch_count) : 20 - 학습률(learning rate) : 0.001 - 은닉 계층 개수 = 3
데이터 정확도	도메인 : 86.4%, 상품 : 24.6%
소요 시간	130s

결과 화면	<pre>om1 = AdamModel('office31_model_1', od, [64,32,10]) om1.exec_all(epoch_count=20, report=10)</pre> <p>Model office31_model_1 train started: Epoch 10: cost=3.230, accuracy=0.829+0.194/0.860+0.120 (65/65 secs) Epoch 20: cost=2.880, accuracy=0.863+0.273/0.830+0.170 (65/130 secs) Model office31_model_1 train ended in 130 secs: Model office31_model_1 test report: accuracy = 0.864+0.246, (0 secs)</p> <p>Model office31_model_1 Visualization</p>  <p>[도메인 추정결과] 추정확률분포 [61,17,22] => 추정 amazon : 정답 amazon => 0 추정확률분포 [100, 0, 0] => 추정 amazon : 정답 amazon => 0 추정확률분포 [84, 5,11] => 추정 amazon : 정답 amazon => 0 [상품 추정결과] 추정확률분포 [4, 1, 3, 4, 3, 4, 3, 3,...] => 추정 bookcase : 정답 back_pack => X 추정확률분포 [0, 1, 2, 1, 0, 1, 0, 0,...] => 추정 stapler : 정답 letter_tray => X 추정확률분포 [1, 1, 1, 6, 6, 6, 4, 8,...] => 추정 desk_lamp : 정답 ring_binder => X</p>
-------	--

(4) 모델4

구분	내용
조건	<ul style="list-style-type: none"> - 에포크 횟수(epoch_count) : 50 - 학습률(learning rate) : 0.001 - 은닉 계층 개수 = 3
데이터 정확도	도메인 : 85.9%, 상품 : 29.1%
소요 시간	345s

결과 화면	<pre> om2 = AdamModel('office31_model_2', od, [64,32,10]) om2.exec_all(epoch_count=50, report=10, learning_rate=0.001) Model office31_model_2 train started: Epoch 10: cost=3.221, accuracy=0.854+0.180/0.830+0.170 (67/67 secs) Epoch 20: cost=2.909, accuracy=0.880+0.252/0.880+0.140 (69/136 secs) Epoch 30: cost=2.634, accuracy=0.891+0.309/0.860+0.260 (70/206 secs) Epoch 40: cost=2.457, accuracy=0.895+0.353/0.890+0.240 (70/276 secs) Epoch 50: cost=2.350, accuracy=0.907+0.373/0.890+0.260 (69/345 secs) Model office31_model_2 train ended in 345 secs: Model office31_model_2 test report: accuracy = 0.859+0.291, (0 secs) Model office31_model_2 Visualization </pre> <div style="display: flex; justify-content: space-around; align-items: center;">    </div> <pre> [도메인 추정결과] 추정확률문포 [100, 0, 0] => 추정 amazon : 정답 amazon => 0 추정확률문포 [74, 7,19] => 추정 amazon : 정답 dslr => X 추정확률문포 [100, 0, 0] => 추정 amazon : 정답 amazon => 0 [상품 추정결과] 추정확률문포 [0, 0, 6, 0, 0, 0, 0, 0,...] => 추정 letter_tray : 정답 ring_binder => X 추정확률문포 [0, 0, 0, 2,14, 6, 1, 3,...] => 추정 bottle : 정답 stapler => X 추정확률문포 [3, 0, 1, 9, 1,16, 0, 1,...] => 추정 calculator : 정답 desktop_computer => X </pre>
-------	--

(5) 모델5

구분	내용
조건	<ul style="list-style-type: none"> - 에포크 횟수(epoch_count) : 20 - 학습률(learning rate) : 0.0001 - 은닉 계층 개수 = 3
데이터 정확도	도메인 : 80.5%, 상품 : 19.0%
소요 시간	135s

결과 화면	<pre>om1 = AdamModel('office31_model_1', od, [64,32,10]) om1.exec_all(epoch_count=20, report=10, learning_rate=0.0001)</pre> <p>Model office31_model_1 train started: Epoch 10: cost=3.663, accuracy=0.814+0.105/0.800+0.080 (67/67 secs) Epoch 20: cost=3.298, accuracy=0.840+0.189/0.840+0.150 (68/135 secs) Model office31_model_1 train ended in 135 secs: Model office31_model_1 test report: accuracy = 0.805+0.190, (0 secs)</p> <p>Model office31_model_1 Visualization</p>  <p>[도메인 추정결과] 추정 확률분포 [54,28,18] => 추정 amazon : 정답 amazon => 0 추정 확률분포 [83, 3,14] => 추정 amazon : 정답 amazon => 0 추정 확률분포 [63,15,22] => 추정 amazon : 정답 dslr => X [상품 추정결과] 추정 확률분포 [9, 0, 1, 4, 2, 6, 3, 2,...] => 추정 mobile_phone : 정답 mobile_phone => 0 추정 확률분포 [1, 0, 7, 0, 0, 0, 0, 0,...] => 추정 projector : 정답 projector => 0 추정 확률분포 [1, 2, 3, 1, 1, 1, 2, 1,...] => 추정 projector : 정답 stapler => X</p>
-------	--

2.2 아담 알고리즘을 사용한 경우


(1) 모델 6

구분	내용
조건	<ul style="list-style-type: none"> - 에포크 횟수(epoch_count) : 20 - 학습률(learning rate) : 0.001 - 은닉 계층 개수 = 3
데이터 정확도	도메인 : 84.5%, 상품 : 22.7%
소요 시간	133s

결과 화면	<pre>om3 = AdamModel('office31_model_3', od, [64,32,10]) om2.use_adam = True om3.exec_all(epoch_count=20, report=10, learning_rate=0.001)</pre> <p>Model office31_model_3 train started: Epoch 10: cost=3.182, accuracy=0.856+0.197/0.930+0.130 (61/61 secs) Epoch 20: cost=2.860, accuracy=0.880+0.259/0.850+0.200 (72/133 secs) Model office31_model_3 train ended in 133 secs: Model office31_model_3 test report: accuracy = 0.845+0.227, (0 secs)</p> <p>Model office31_model_3 Visualization</p>  <p>[도메인 추정결과] 추정확률분포 [92, 3, 5] => 추정 amazon : 정답 amazon => 0 추정확률분포 [100, 0, 0] => 추정 amazon : 정답 amazon => 0 추정확률분포 [79,18, 3] => 추정 amazon : 정답 amazon => 0 [상품 추정결과] 추정확률분포 [0, 1, 1, 0, 0, 0, 0, 0,...] => 추정 keyboard : 정답 mug => X 추정확률분포 [0, 1, 0, 9, 3, 4,41, 9,...] => 추정 desk_chair : 정답 desktop_computer => X 추정확률분포 [2, 0, 4, 0, 1, 6, 0, 0,...] => 추정 mouse : 정답 pen => X</p>
-------	---




(2) 모델 7

구분	내용
조건	<ul style="list-style-type: none"> - 에포크 횟수(epoch_count) : 50 - 학습률(learning rate) : 0.001 - 은닉 계층 개수 = 3
데이터 정확도	도메인 : 85.6%, 상품 : 25.6%
소요 시간	343s

결과 화면	<pre>om3 = AdamModel('office31_model_3', od, [64,32,10]) om2.use_adam = True om3.exec_all(epoch_count=50, report=10, learning_rate=0.001)</pre> <p>Model office31_model_3 train started: Epoch 10: cost=3.302, accuracy=0.816+0.164/0.860+0.190 (68/68 secs) Epoch 20: cost=2.916, accuracy=0.861+0.240/0.840+0.170 (69/137 secs) Epoch 30: cost=2.667, accuracy=0.891+0.302/0.880+0.180 (70/207 secs) Epoch 40: cost=2.491, accuracy=0.900+0.345/0.890+0.180 (68/275 secs) Epoch 50: cost=2.453, accuracy=0.902+0.349/0.880+0.230 (68/343 secs) Model office31_model_3 train ended in 343 secs: Model office31_model_3 test report: accuracy = 0.856+0.256, (0 secs)</p> <p>Model office31_model_3 Visualization</p>  <p>[도메인 추정결과] 추정 확률분포 [31,14,55] => 추정 webcam : 정답 webcam => 0 추정 확률분포 [97, 1, 2] => 추정 amazon : 정답 amazon => 0 추정 확률분포 [97, 1, 2] => 추정 amazon : 정답 amazon => 0 [상품 추정결과] 추정 확률분포 [0, 0, 0, 1,11, 8, 0, 7,...] => 추정 mobile_phone : 정답 bottle => X 추정 확률분포 [0, 0, 0, 0, 0, 1, 0, 0,...] => 추정 projector : 정답 printer => X 추정 확률분포 [0, 0, 2, 1, 1, 2, 1, 1,...] => 추정 projector : 정답 ring_binder => X</p>
-------	--

(3) 모델 8

구분	내용
조건	<ul style="list-style-type: none"> - 에포크 횟수(epoch_count) : 20 - 학습률(learning rate) : 0.0001 - 은닉 계층 개수 = 3
데이터 정확도	도메인 : 82.1%, 상품 : 17.0%
소요 시간	138s

결과 화면	<pre> om3 = AdamModel('office31_model_3', od, [64,32,10]) om2.use_adam = True om3.exec_all(epoch_count=20, report=10, learning_rate=0.0001) Model office31_model_3 train started: Epoch 10: cost=3.614, accuracy=0.799+0.127/0.850+0.140 (68/68 secs) Epoch 20: cost=3.332, accuracy=0.831+0.179/0.850+0.150 (70/138 secs) Model office31_model_3 train ended in 138 secs: Model office31_model_3 test report: accuracy = 0.821+0.170, (0 secs) Model office31_model_3 Visualization </pre> <div>    </div> <pre> [도메인 추정결과] 추정확률분포 [96, 1, 3] => 추정 amazon : 정답 amazon => 0 추정확률분포 [23,27,50] => 추정 webcam : 정답 amazon => X 추정확률분포 [86, 2,11] => 추정 amazon : 정답 amazon => 0 [상품 추정결과] 추정확률분포 [0, 7, 1, 1, 2, 2, 8, 4,...] => 추정 scissors : 정답 back_pack => X 추정확률분포 [2, 1, 4, 0, 0, 0, 1, 0,...] => 추정 laptop_computer : 정답 desktop_computer => X 추정확률분포 [0,13, 1, 1, 1, 2, 5, 4,...] => 추정 bike : 정답 stapler => X </pre>
-------	---

3. 결론

실험 결과를 표와 함께 정리하면 다음과 같다.

(1) 아담 알고리즘을 사용하지 않은 경우

	모델 1	모델 2	모델 3	모델 4	모델 5
은닉 계층	[10]	[10]	[64, 32, 10]	[64, 32, 10]	[64, 32, 10]
은닉 계층 수	1	1	3	3	3
에포크 횟수	20	50	20	50	20
학습률	0.001	0.001	0.001	0.001	0.0001
도메인 정확도(%)	66.1	66.1	86.4	85.9	80.5
품목 정확도(%)	4.8	4.8	24.6	29.1	19.0
소요 시간(s)	68	155	130	345	135

표를 보면 은닉계층 수가 1일 때, 도메인 정확도 및 품목 정확도가 약 66%, 5% 정도로 은닉계층 수가 3일 때에 비해 약 15%, 20% 낮은 모습을 볼 수 있다. 또한, 에포크 횟수

가 다름에도 불구하고 66.1%, 4.8%로 똑같이 나타난 것으로 보아 일정한 추정 확률 분포에서 벗어나지 못한다는 것을 알 수 있다. 즉, 이를 통해 학습이 미비함을 유추해볼 수 있는 것이다. 따라서 이후에는 은닉 계층 수가 3개인 [64, 32, 10]을 사용해 실험을 진행하였다.

은닉계층 [64,32,10] 모델에서 에포크 횟수를 늘려 실험해 본 결과, 도메인 정확도는 86.4%에서 85.9%로 소폭 하락했지만, 품목 정확도는 24.6%에서 29.1%로 증가했음을 알 수 있었다. 특히, 소요시간이 130초에서 345초로 크게 차이났다. 학습률을 0.0001로 변경해 본 결과, 도메인 정확도는 86.4%에서 80.5%로 감소하고, 품목 정확도는 24.6%에서 19.0%으로 하락했다. 이 결과를 통해 에포크 횟수 20, 학습률 0.001의 조건에서 좋은 결과가 도출됨을 알 수 있다. 중요한 점은, 은닉 계층 1개를 사용할 때와 달리 추정 확률 분포의 데이터별 차별화를 확인 할 수 있으며, 도메인과 품목 모두에서 정확도가 향상되었다는 점이다.

(2) 아담 알고리즘을 사용한 경우

	모델 3	모델 4	모델 5	모델 6	모델 7	모델 8
은닉 계층	[64,32,10]	[64,32,10]	[64,32,10]	[64,32,10]	[64,32,10]	[64,32,10]
은닉 계층 수	3	3	3	3	3	3
에포크 횟수	20	50	20	20	50	20
학습률	0.001	0.001	0.0001	0.001	0.001	0.0001
도메인정확도(%)	86.4	85.9	80.5	84.5	85.6	82.1
품목 정확도(%)	24.6	29.1	19.0	22.7	25.6	17.0
소요 시간(s)	130	345	135	133	343	138

위 표에서는 (1)에서 사용한 은닉계층[64,32,10] 모델 (아담 알고리즘을 사용하지 않음)과 아담 알고리즘을 사용한 모델을 비교해 나타내었다. 모델 3과 모델 6을 보면, 도메인 정확도가 86.4%, 84.5%로 아담 알고리즘을 사용했을 때가 약간 낮게 나타났고, 품목 정확도

또한 24.6%, 22.7%로 아담 알고리즘 이용 시 낮게 나타났다. 하지만 미비한 차이를 보여 비슷한 결과를 나타내었다고 볼 수 있다. 마찬가지로 모델 4와 모델 7, 모델 5와 모델 8의 경우에서도 아담 알고리즘 사용 경우의 정확도가 조금 더 낮게 나타났다. 하지만 소요시간의 경우, 에포크 횟수가 50일 때, 20인 경우와 다르게 아담 알고리즘을 이용한 경우 소요시간이 2초 적게 사용되었음을 확인할 수 있었다. 이를 통해 아담 알고리즘이, 학습량이 많을 때 빠른 수행이 가능함을 유추해 볼 수 있다.

3-1. 추가 분석

(1) Stochastic Gradient 방법과 아담 알고리즘의 차이

Stochastic Gradient Descent는 확률적 경사 하강법으로, 딥러닝 모델의 학습에서 가장 기본적으로 사용되는 최적화 알고리즘 중 하나이다. 경사 하강법의 확률적인 버전으로, 각 스텝마다 무작위로 선택된 미니배치에 대한 손실함수의 기울기를 이용해 모델을 업데이트한다.

아담 알고리즘은 현재 가장 널리 사용되는 최적화 알고리즘으로, 이전 기울기의 지수적인 이동 평균을 사용하여 학습률을 조절하는 방식으로 모델을 업데이트한다. 경사하강법 알고리즘을 기반으로 하면서도, 모멘텀 및 학습률 감소와 같은 개선된 기능을 추가한 최적화 알고리즘이다.

SGD(Stochastic Gradient Descent)는 미니배치를 통해 학습시키는 경우 최적의 값을 찾아가기 위한 방향 설정이 뒤죽박죽이고, 한 스텝을 나아가기 위한 사이즈를 정하기가 어렵다.¹ 이와 다르게 방향과 스텝 사이즈를 고려한 optimizer가 아담 알고리즘이라 할 수 있다.

즉, SGD와 아담 알고리즘은 모두 경사하강법을 기반으로 하지만, 학습 과정에서 최적의 값

¹ 한 걸음 나아가기 위한 보폭이 낮으면 학습하는데 오래 걸리고, 너무 크면 최적의 값을 찾지 못하는 문제가 있다.

을 찾아가는 방향 및 스텝 사이즈에서 차이가 나타난다.

(2) 전이 학습의 필요성

전이 학습이란 어떤 목적을 이루기 위해 학습된 모델을 다른 작업에 이용하는 것을 말한다. 전이 학습 사용 시 데이터셋의 크기와 상관 없이 장점을 가진다. 우선 큰 데이터셋을 이용하는 경우, 학습이 빠르게 수행될 수 있다. 이미 입력되는 데이터에 대해 특징을 효율적으로 추출하기에 특징 추출을 위한 학습은 할 필요가 없기 때문이다. 반대로 작은 데이터셋에 대해 학습할 때는 오버피팅을 예방할 수 있다. 적은 데이터로 특징 추출 학습을 하면, 데이터 수 대비 모델의 가중치 수가 많아 미세한 특징까지 모두 학습하는 문제가 생길 수 있다. 그런데 전이 학습을 이용해 마지막 레이어만 학습하게 하면, 학습할 가중치를 줄일 수 있어 과한 학습을 방지할 수 있게 된다. 따라서 전이 학습은 데이터셋의 크기와 상관 없이 강점을 가질 수 있는 학습 방법이라 할 수 있다.

cf) 전이 학습이 잘 되기 위한 조건: 사전 학습에 사용한 데이터와 새로운 데이터가 비슷한 형태를 가지고, 새로 학습할 데이터에도 비슷한 특징을 활용할 수 있어야 한다. 크게 다른 형태의 데이터를 사전 학습된 모델에 적용한다고 해도 특징의 재추출이 필요할 것이기 때문이다. 또한, 일반적으로 새로운 데이터보다 많은 데이터로 사전 학습이 수행되었어야 한다. 새로운 데이터가 사전 학습에 사용된 데이터보다 많으면, 해당 지식의 의미 자체가 없어질 수 있기 때문이다.