Multi-Task Learning에서 공유 공간과 성능과의 관계 탐구

성수진[O1], 박성재[1], 정인규[2], 차정원[1]
창원대학교[1], 치안정책연구소[2]
{20153057, tjdwo1289, jcha}@changwon.ac.kr[1], ikjeong@police.go.kr[2]

Exploring the Relationship between Shared Space and Performance in Multi-Task Learning

Su-Jin Seong[O1], Seong-Jae Park[1], In-Gyu Jeong[2], Jeong-Won Cha[1]
Changwon National University[1], Police Science Insitute[2]

요 약

딥러닝에서 층을 공유하여 작업에 따라 변하지 않는 정보를 사용하는 multi-task learning이 다양한 자연어 처리 문제에 훌륭하게 사용되었다. 그렇지만 우리가 아는 한 공유 공간의 상태와 성능과의 관계를 조사한 연구는 없었다. 본 연구에서는 공유 공간과 task 의존 공간의 자질의 수와 오염 정도가 성능에 미치는 영향도 조사하여 공유 공간과 성능 관계에 대해서 탐구한다. 이 결과는 multi-task를 진행하는 실험에서 공유 공간의 역할과 성능의 관계를 밝혀서 시스템의 성능 향상에 도움이 될 것이다.

주제어: 문서 분류, multi-task learning, adversarial learning

1. 서론

multi-task learning은 single-task learning의 성능을 올리는데 효과적인 방법이다. 인공 신경망을 이용한 multi-task learning은 컴퓨터 비전[1, 2], 자연어처리[3, 4]에서 우수한 결과를 보여 주었다.

그렇지만 multi-task learning[5]에 대한 대부분의 이전 연구는 자질이 공유 공간에 있어야 하는지 아닌지에 따라 자질을 나누는 것이었다. 그림1에서 보는 것과 같이 task에 의존적인 자질과 공유 자질로 나누어지게 된다. 이러한 단순 공유 공간의 문제점은 불필요한 task 의존적인 자질이 함께 자리하게 되는 것이다.

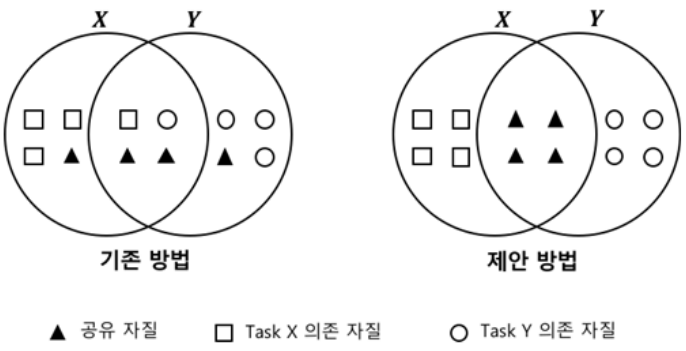

그림 1. task X, task Y에서의 multi-task 방법

그림 1은 task X, task Y에서의 multi-task 방법으로 □는 task X 의존 자질이고 ○은 task Y 의존적인 자질이다. ▲은 공유자질이다. 왼쪽 그림은 자질들이 오염되어 있고 오른쪽 공간은 task에 적합한 자질들이 공유 공간과 task 의존 공간에 분포되어 있다.

본 연구에서는 multi-task 학습에서 공유 공간과 task 의존 공간의 오염과 성능과의 관계에 대해 다음과 같은 가설을 설정한다.

가설: task 의존 공간과 공유 공간의 오염이 적을수록 분류 성능이 향상된다.

이러한 가설을 확인하기 위해서 각 공간과 성능과의 관계를 확인한다.

본 연구의 기여점은 다음과 같다.

- multi-task 딥러닝의 분류 문제에서 공유 공간과 task 의존 공간에서 성능 향상을 위해 자질들의 분포를 확인했다.
- 확인된 공유 공간의 특징을 이용하여 다른 문제에서도 이 지식을 활용하여 성능 향상을 꾀할 수 있을 것으로 본다.

2. 관련 연구

multi-task learning은 별개의 task를 동시에 학습하

여 서로의 정보를 부분적으로 공유하며 학습의 효율을 높이는 방법이다.

문장의 의미를 파악하기 위해 많은 연구들이 있었다. [8, 9, 10]에서는 RNN을 이용하여 모델링 하였고 [11, 12]은 CNN을 이용하여 문장을 모델링하였다. [13]은 Recursive NN을 이용하여 모델링하였다. 우리는 CNN을 문장 모델링을 위해 채택하였다.

[6]은 한국어 의존 구문 분석을 위하여 RNN에 Attention mechanism을 추가한 포인트 네트워크로 두 어절 간의 의존 관계와 의존 레이블 정보를 분석하는 multi-task learning을 수행하였다. 이를 통해 기존 의존 구문 분석 모델들을 적용하지 않고 의존 관계를 파악할 수 있었고 기존 연구 성능인 UAS(Unlabeled Attachment Score) 88~90%보다 높은 91.3 ~ 01.65%의 성능을 보였다.

[7]은 한국어 개체명 인식을 위하여 multi-task Highway Bi-LSTM-CRFs 모델을 제안하였고, 개체명 인식과 양방향 언어 모델의 학습을 동시에 진행하여 별도의 레이블링 작업 없이 언어 모델에서 학습할 수 있는 한국어의 언어적 특성을 개체명 인식 모델이 부분적으로 이용하도록 하였다. multi-task 학습을 진행한 경우 기본적인 Bi-LSTM 모델을 사용한 경우보다 0.46%의 성능 향상이 있었다.

3. 제안 구조

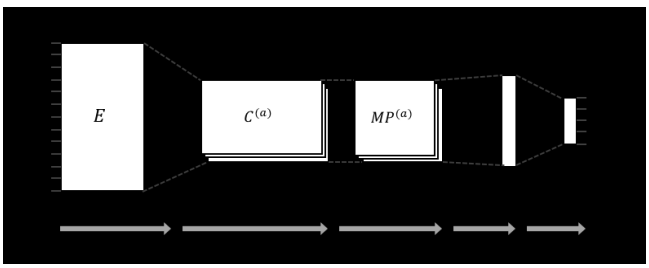

그림 2. Single model

그림 2는 단일 task에 대하여 문장의 카테고리를 분류하는 모델이다. E는 입력에 대한 음절단위 embedding matrix이고 $C^{(a)}$는 task A에 대한 convolution layer, $MP^{(a)}$는 Max-pooling layer이다. $Y^{(a)}$는 분류 범주이다.

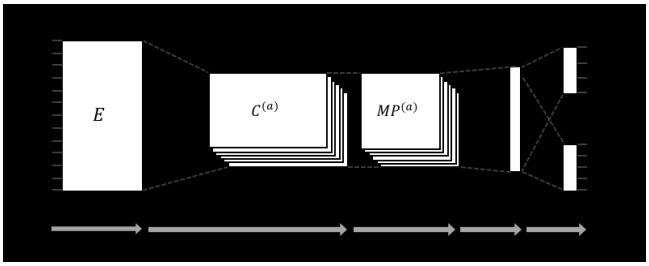

그림 3. Concatenation model

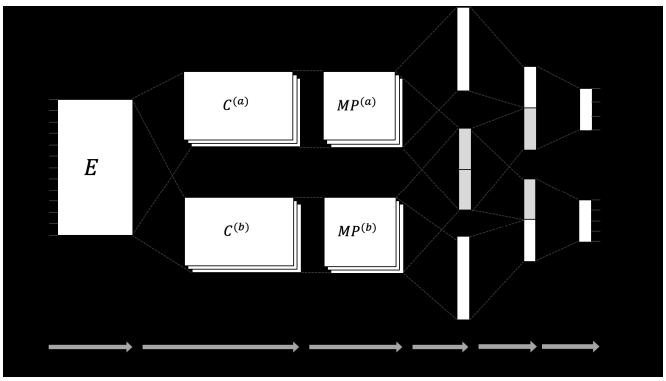

그림 4 Duplication model

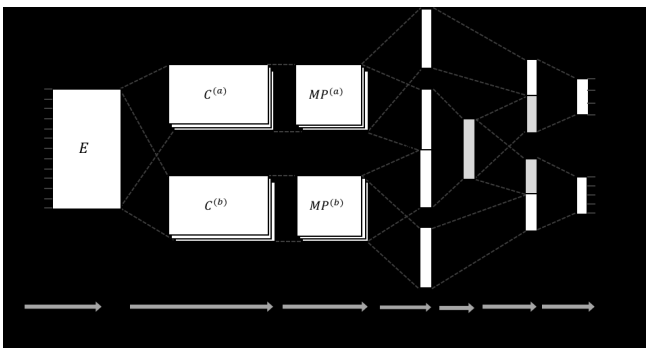

그림 5. Fully Connected model

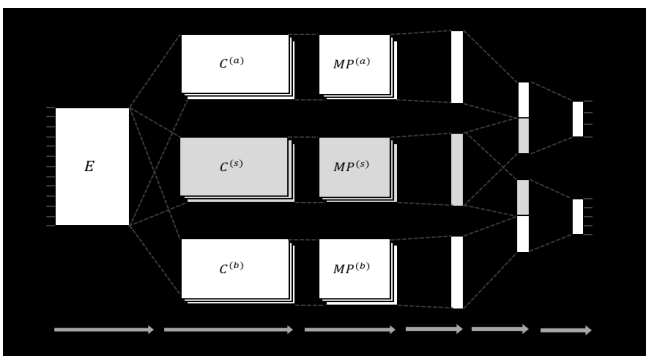

그림 6. Input shared model

그림 3~6은 다양한 multi-task model의 구조이다. 그림 3은 single model에서와 같은 convolution,

max-pooling layer를 두 개를 가지고, 두 개의 task에 대한 분류를 수행한다. 분류해야할 task가 두 개로 늘어나면서 동시에 고려해야할 자질이 증가하였고 이에 맞추어 2배로 늘린 convolution layer가 이를 적절히 추출할 것이라 기대된다.

그림 4는 그림 3의 두 convolution, max-pooling의 결과가 task A의 경우 $2A+B$, task B의 경우 $2B+A$로 제공된다. 이 경우 각 task에 대한 loss가 각각 CNN에 두 배로 영향을 줄 수 있고, 이를 통해 두 개의 CNN 구조는 각 task에 편향된 feature를 추출할 것이다.

그림 5는 그림 3에서 두 가지 loss에 모두 영향을 받는 Fully connected layer를 추가하였다. Fully connected layer는 공유 레이어의 차원을 줄이며 유익한 자질을 걸러내는데 도움이 될 것이라 기대된다.

위 모델들은 두 개로 나누어진 Convolution layer가 두 task의 영향을 모두 받을 수 있다. 그림 6의 모델은 convolution layer가 각 task에 특화되도록 하기 위해 Input으로부터 공통된 feature를 추출할 convolution layer를 추가하고 이 외의 convolution layer는 목표로 하는 task의 영향만 받도록 구축되었다.

4. 실험

4.1. 실험 설정

첫 번째 실험은 '서울지방경찰청 112신고데이터'를 사용하여 사건의 종류와 그 사건의 위험도를 분류하는 문제이다. 여기서 Type은 사건의 종류를 분류하는 것이고 Risk는 각 사건에서 위험도를 분류하는 것이다. 사건의 종류는 A에서 D까지 총 4 가지이고 평가 데이터 총 3,645개 중 D가 639개로 전체 66.04%를 차지한다. Risk의 종류는 C0에서 C4까지 총 5개이고 C1이 전체 3,645개 중 2,733개로 74.98%의 비율을 가지고 있다. 두 task의 chi-square test결과는 x-squared가 7586.8이고, p-value가 $2.2e^{-16}$보다 작아 통계적으로 연관성이 있다.

두 번째 실험은 고객 VOC 데이터에서 제품과 서비스를 분류하는 실험이다. Product는 제품의 종류를 분류하는 것이고 Event는 서비스의 종류를 분류하는 것이다. Product의 경우 70개의 카테고리 중 가장 빈도가 높은 카테고리가 전체 평가 데이터 5443개 중 2504개로 46%를 차지한다. Event의 경우 27개의 카테고리 중 가장 빈도가 높은 카테고리가 2503개로 46%를 차지한다. Product와 Event의 chi-square test 결과 x-squared가 14253이고, p-value가 $2.2e^{-16}$보다 작아 통계적으로 연관성이 있는 task이다.

데이터의 카테고리의 비율이 불균형하기 때문에 모든 입력에 대해 비율이 높은 카테고리로만 예측하더라도 높은 Accuracy를 얻을 수 있다. 모델이 소수의 카테고리도 잘 예측하는지를 반영하기 위해 성능은 precision, recall, F1-score의 Macro average와 Accuracy로 측정하여 표기한다. 식 1은 Accuracy, 식 2는 Macro average precision, 식 3은 Macro average recall, 식 4는 Macro F1-score를 나타낸다. Accuracy는 Micro precision, Micro recall과 동일하다.

$$Accuracy = \frac{\sum_{i}^{c}(TP_i + TN_i)}{\sum_{i}^{c}(TP_i + TN_i + FP_i + FN_i)} \quad (1)$$

$$Precision = \frac{\sum_{i}^{c} TP_i}{\sum_{i}^{c} TP_i + \sum_{i}^{c} FP_i} \quad (2)$$

$$Recall = \frac{\sum_{i}^{c} TP_i}{\sum_{i}^{c} TP_i + \sum_{i}^{c} FN_i} \quad (3)$$

$$F1-score = \frac{\sum_{i}^{c}(\frac{2 \times Precision_i \times Recall_i}{Precision_i + Recall_i})}{c} \quad (4)$$

이 때 c는 task에 포함된 카테고리의 수를 나타낸다.

4.2. 실험 결과 및 분석

실험에 사용한 2개의 데이터 쌍은 각각 카이제곱검정을 수행한 결과 p-value가 0.05 이하의 값을 가졌고, 이는 두 task가 통계적으로 연관이 있다는 것을 의미한다.

전체 실험에서 embedding size는 50, 필터 size는 3, 4, 5로 설정하였으며 각 필터는 10개씩 사용하였다. 또한 dropout을 적용하였고, optimizer로 adam을 사용하였다.

표 1~4는 task들에 대한 성능표이다. Baseline은 모든 입력에 대해 빈도가 가장 높은 카테고리로만 예측하였을 경우로 설정하여 성능을 측정한 결과이다.

표 1 Type의 성능 결과표

	precision	recall	F1	accuracy
Baseline	0.165	0.250	0.199	0.660
Single	0.874	0.821	0.845	0.929
Concatenation	0.881	0.803	0.836	0.930
Duplication	0.889	0.845	**0.866**	**0.933**
Fully Connected	0.858	0.853	0.854	0.926
Input shared	0.871	0.829	0.849	0.932

표 2 Risk의 성능 결과표

	precision	recall	F1	accuracy
Baseline	0.150	0.200	0.171	0.750
Single	0.413	0.235	0.240	0.751
Concatenation	0.386	0.240	0.243	0.753
Duplication	0.545	0.295	**0.320**	0.753
Fully Connected	0.631	0.265	0.286	**0.757**
Input shared	0.466	0.253	0.262	**0.757**

표 3 Product의 성능 결과표

	precision	recall	F1	accuracy
Baseline	0.007	0.014	0.009	0.460
Single	0.914	0.844	0.864	0.961
Concatenation	0.665	0.477	0.518	0.902
Duplication	0.951	0.859	**0.891**	**0.967**
Fully Connected	0.787	0.672	0.706	0.947
Input shared	0.759	0.663	0.688	0.942

표 4 Event의 성능 결과표

	precision	recall	F1	accuracy
Baseline	0.017	0.037	0.023	0.460
Single	0.882	0.742	0.786	0.970
Concatenation	0.940	0.821	0.863	0.973
Duplication	0.974	0.900	**0.930**	**0.982**
Fully Connected	0.953	0.915	0.929	**0.982**
Input shared	0.935	0.849	0.884	0.979

실험 결과 모든 multi-task모델이 baseline에 비해서 성능이 개선되었다. Duplication 모델의 경우에 Macro 성능이 가장 우수했다. 두 실험이 모두 연관 있는 데이터를 사용하였고 동일한 입력을 가지기 때문에 공유되는 자질이 많을 것이라고 판단된다. 이를 확인하기 위해 평가 데이터셋에 대한 tri-gram을 생성하고 이를 입력으로 모델의 마지막 concatenation layer들의 결과를 추출한 후 결과의 합이 큰 상위 6000개를 추출하였다. concatenation layer의 결과는 softmax를 거쳐 카테고리를 결정하기 때문에 결과 값이 클수록 각 분류에 대한 영향력도 크다. 추출된 6000개의 결과에 나타나는 단어의 빈도수를 표 5, 6에 나타낸다.

표 5 Risk와 Type task에서 공유 공간과 task 의존 공간에서 자질들의 수

	Risk	Shared	Type
Single	4902	1098	4902
Concatenation	4672	1328	4672
Duplication	**1784**	**4216**	**1784**
FC shared	1717	4283	1717
Input shared	3293	2707	3293

표 6 Event와 Product task에서 공유 공간과 task 의존 공간에서 자질들의 수

	Event	Shared	Product
Single	4138	1862	4138
Concatenation	2338	3662	2338
Duplication	**1867**	**4133**	**1867**
FC shared	2838	3162	2838
Input shared	3427	2573	3427

공유되지 않는 의존 자질은 공유 자질보다 분류에 더 큰 영향력을 가지기 때문에, 각 task에 적합하지 않는 자질이 해당 task 의존 공간에 존재한다면 이는 노이즈로 작용할 수 있다. 학습 결과를 확인했을 때 전반적으로 공유되는 자질이 많은 경우 즉, task 의존 공간에 존재하는 의존 자질이 적은 경우 좋은 성능을 보임을 알 수 있었다. 이러한 경향을 보아 의존 자질이 적은 경우 의존 공간의 오염도가 감소한다고 추측된다. 각 task는 서로 상관없는 task가 아니라 상관관계가 큰 task이기 때문에 공통적으로 영향력이 높은 자질이 많고 하나의 task에만 유의하고 상대 task에 혼란을 주는 자질이 적다. 따라서 공유 공간보다 의존 공간의 자질이 많은 경우 오염도가 높다고 볼 수 있다.

5. 결론

본 논문에서는 multi-task 학습에서 공유 공간과 task 의존 공간에서 각 자질들이 성능에 미치는 영향을

조사하였다.

실험 결과 task 의존 공간 내의 자질 수가 적을수록 성능이 향상되는 경향이 있었다. 본 연구에 사용된 task는 상관관계가 높은 task로 task에 의존적인 자질이 공통적으로 유의한 자질보다 적기에 의존 공간 내의 자질의 수가 많으면 의존 공간의 오염도가 높아진다고 볼 수 있다. 즉 의존 공간의 자질 수가 적을수록 오염될 확률이 낮다. 따라서 각 공간의 오염이 적을수록 성능이 향상될 것이라는 우리의 가설을 확인하였으며 향후에는 오염을 줄일 수 있는 학습법에 대해서 연구를 계속할 것이다.

Acknowledgement

이 논문은 2018년도 정부(과학기술정보통신부)의 재원으로 정보통신기술진흥센터의 지원을 받아수행된 연구임 (No.2018-0-00440, 위험 상황 초기 인지를 위한 ICT 기반의 범죄 위험도예측 및 대응 기술 개발)

참고문헌

[1] Ishan Misra, Abhinav Shrivastava, Abhinav Gupta, Martial Herbert, “Cross-Stitch Networks for Multi-Task Learning”, The IEEE Conference on Computer Vision and Pattern Recognition(CVPR), pp.3994-4003, 2016

[2] Zhanpeng Zhang, Ping Luo, Chen Change Loy, and Xiaoou Tang. 2014. Facial landmark detection by deep multi-task learning. In European Conference on Computer Vision. Springer, pages 94-108.

[3] Ronan Collobert, Jason Weston, “A Unified Architecture for Natural Language Processing: Deep Neural Netwroks with Mutitask Learning”, The 25th International Conference on Machine Learning. ACM, pp.160-167, 2008

[4] Minh-Thang Luong, Quoc V Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. 2015. Multi-task sequence to sequence learning. arXiv preprint arXiv:1511.06114 .

[5] Pengfei Liu, Xipeng Qiu, Xuanjing Huang, “Deep Multi-Task Learning with Shared Memory”, arXiv preprint arXiv:1609.07222, 2016

[6] 박천음, 이창기, “멀티 태스크 학습 기반 포인터 네트워크를 이용한 한국어 의존 구문 분석”, 한국정보과학회 학술발표논문집, pp.440-442, 2016

[7] 박찬민, 김병재, 서정연, “Highway Bi-LSTM-CRFs 모델을 이용한 멀티 태스크 기반 한국어 개체명 인식”, 한국HCI학회 학술대회, pp.432-435, 2018

[8] Ilya Sutskever, Oriol Vinyals, and Quoc VV Le. 2014. Sequence to sequence learning with neural networks. In Advances in NIPS. pages 3104-3112.

[9] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555 .

[10] PengFei Liu, Xipeng Qiu, Xinchi Chen, Shiyu Wu, and Xuanjing Huang. 2015. Multi-timescale long short-term memory neural network for modelling sentences and documents. In Proceedings of the Conference on EMNLP.

[11] Ronan Collobert, Jason Weston, L´eon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. The JMLR 12:2493-2537.

[12] Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A convolutional neural network for modelling sentences. In Proceedings of ACL.

[13] Richard Socher, Alex Perelygin, Jean Y Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng,and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In Proceedings of EMNLP.