The Korean Journal of Applied Statistics (2019) DOI: https://doi.org/10.5351/KJAS.2019.32.1.041
32(1), 41–68

Multi-view learning review: understanding methods and their application

Kang Il Bae[a] · Yung Seop Lee[b] · Changwon Lim[a,1]

[a]Department of Applied Statistics, Chung-Ang University;
[b]Department of Statistics, Dongguk University

(Received November 6, 2018; Revised December 13, 2018; Accepted January 3, 2019)

Abstract

Multi-view learning considers data from various viewpoints as well as attempts to integrate various information from data. Multi-view learning has been studied recently and has showed superior performance to a model learned from only a single view. With the introduction of deep learning techniques to a multi-view learning approach, it has showed good results in various fields such as image, text, voice, and video. In this study, we introduce how multi-view learning methods solve various problems faced in human behavior recognition, medical areas, information retrieval and facial expression recognition. In addition, we review data integration principles of multi-view learning methods by classifying traditional multi-view learning methods into data integration, classifiers integration, and representation integration. Finally, we examine how CNN, RNN, RBM, Autoencoder, and GAN, which are commonly used among various deep learning methods, are applied to multi-view learning algorithms. We categorize CNN and RNN-based learning methods as supervised learning, and RBM, Autoencoder, and GAN-based learning methods as unsupervised learning.

Keywords: multi-view learning, multi-modal learning, deep learning, machine learning, data integration

1. 서론

인간은 어떤 현상을 학습할 때 다양한 유형의 데이터를 이용한다. 예를 들어, 영화를 볼 때 화면에 나오는 영상뿐만 아니라 영화에서 나오는 음성이나 자막을 함께 이용하면 더욱 쉽게 영화를 이해할 수 있다. 최근 기계 학습 분야에서 중요한 트렌드로 떠오르고 있는 멀티 뷰(multi-view) 학습은 인간의 인지적 학습 방법을 모방하여 다양한 뷰의 데이터로부터 학습하는 방법이다. 멀티 뷰 학습의 목표는 멀티 뷰 데이터의 다양한 정보를 이용하여 단일 뷰의 데이터를 사용했을 때보다 학습의 성능을 높이는 것이다 (Zhao 등, 2017).

뷰(view)란 데이터를 보는 관점으로 데이터가 갖는 특징(feature)라고 할 수 있다. 멀티 뷰 데이터는 데이터 안에 존재하는 음성이나 텍스트 등 다른 여러 가지 특징 혹은 다양한 소스로부터 획득한 데이터의

This research was supported by Next-Generation Information Computing Development Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT (NRF-2017 M3C4A7083281).
[1]Corresponding author: Department of Applied Statistics, Chung-Ang University, 84 Heukseok-ro, Dongjak-gu, Seoul 06974, Korea. E-mail: clim@cau.ac.kr

여러 가지 특징을 말한다 (Xu 등, 2013). 멀티 뷰 학습을 위해 사용되는 멀티 뷰 데이터는 접하기 어려운 특별한 데이터가 아닌 일상생활 속에서 쉽게 접할 수 있는 데이터이다. 예를 들어, 문서의 경우 문서 안의 내용 이외에도 관련 저자 등 다양한 데이터의 뷰가 존재하고, social network service (SNS) 데이터의 경우 이용자가 업로드한 사진 이외에 사진을 설명하는 텍스트, 해시 태그(hash tag) 등 다양한 뷰가 존재한다.

멀티 뷰 기법은 다양한 분야에서 연구되고 있고 멀티 뷰 학습의 방법론은 딥 러닝(deep learning) 기법의 도입으로 새로운 변화를 맞이하였다. 딥 러닝은 뇌의 인지 구조를 모방한 기계학습 모형으로 사람이 학습할 때 뇌에서 시냅스(synapse)를 통해 뉴런(neuron)끼리 신호를 주고받는 것처럼 인공신경망은 입력층의 노드와 은닉층의 노드에 연결된 가중치를 조절하는 방식으로 학습한다 (Schmidhuber, 2015). 인공신경망은 데이터의 복잡한 특징을 고려하여 데이터의 표현을 학습할 수 있어 최근 인공지능, 이미지 처리, 번역, 의학 등 여러 분야에서 큰 성공을 거두고 있다 (Vargas 등, 2017). 멀티 뷰 기법에서도 딥 러닝 기법을 도입하여 멀티 뷰 기법의 성능을 향상시킨 연구가 많아지고 있다. 하지만 딥 러닝 기법을 응용한 멀티 뷰 기법에 대한 연구가 많아지고 있음에도 불구하고 이를 정리한 연구는 충분하지 않은 상황이다. 기존의 리뷰 논문들을 살펴보면, Xu 등 (2013)은 멀티 뷰 학습의 원리를 중심으로 멀티 뷰 학습 방법론들을 정리하였으나 딥 러닝 기법들에 대해서는 소개하지 않았다. Li 등 (2016)은 멀티 뷰 데이터의 표현을 학습할 수 있는 기법들을 비신경망적인 방법과 신경망적인 방법으로 나누어 리뷰를 하였다. 하지만 데이터의 표현을 학습하는 방법들만 소개하여 코트레이닝(co-training) 등 전통적인 멀티 뷰 학습 기법은 다루지 않았다. 본 연구에서는 멀티 뷰 학습의 연구에 토대가 된 중요한 방법론들을 리뷰하고, 멀티 뷰 학습 기법이 어떤 식으로 사용되는지를 소개하고자 한다. 또한 최근 급격히 발전하고 있는 딥 러닝 기법들을 응용한 멀티 뷰 기법들을 기계학습 방법론에 따라 분류하고, 이 방법들에 대한 이해를 돕고자 한다.

본 논문은 총 6장으로 구성되어 있다. 2장에서는 멀티 뷰 기법이 사용되는 분야에 대해서 다루었으며 멀티 뷰 기법이 해당 분야에서 직면한 문제를 어떻게 해결하고 있는지에 대하여 서술하였다. 3장에서는 멀티 뷰 기법의 일반론을 설명하여 멀티 뷰 학습이 어떤 식으로 이루어지는지에 대하여 서술하였다. 4장과 5장에서는 멀티 뷰 기법에서 사용되는 딥 러닝 기법을 기계 학습적 방법론 방법에 따라 분류하고 모형의 학습 방식에 대하여 서술하였다. 6장은 본 연구의 결론과 추후 연구 방향에 대하여 서술하였다.

2. 멀티 뷰 학습 기법의 사용 분야

본 장에서는 인간 행동 인식(human activity recognition) 분야, 의학 분야, 정보 검색(information retrieval) 분야, 표정 인식(facial expression recognition) 분야와 같은 다양한 분야에서 멀티 뷰 기법이 어떤 방식으로 연구되고 있는지 살펴볼 것이다.

2.1. 인간 행동 인식 분야

인간 행동 인식 분야는 인간의 복잡한 행동 양식을 이해하고자 하는 것을 목적으로 한다. 인간의 행동은 개인의 성향, 심리적인 상태, 인간의 자아를 반영하는 정보를 갖고 있기 때문에 분석이 쉽지 않다. 또한 영상 안에서는 공간적인 정보와 시간적인 정보가 담겨 있는데 공간적인 정보의 경우 영상마다 객체의 위치, 음영, 위치, 스케일 등이 달라지고 시간적인 정보의 경우 이러한 영상이 시간의 흐름에 따라 달라진다. 이와 같은 시공간적인 정보를 동시에 고려하여 학습을 하는 것은 쉽지가 않다 (Vrigkas 등, 2015).

이러한 문제를 해결하기 위하여 여러 가지 멀티 뷰 기법들이 제안되었는데, Neverova 등 (2016)은 convolutional neural network (CNN)를 이용하여 영상에서 몸짓의 국소 공간적인 특징, 시간에 따른 역

동적인 몸짓을 학습 시키는 방법을 제안하였다. 이때 신경망 모형의 예측이 결측된 레이블이 많은 경우에 대하여도 로버스트한 ModDrop 학습 방식을 제안하여 영상의 몸짓을 분류 예측하였다. Jain 등 (2016)은 structural recurrent neural networks (S-RNN) 모형을 제시하여 RNN 모형을 통하여 영상의 시간적 특징을 학습하고 몸의 여러 부위를 구조화하여 공간적 특징을 고려하였다. 이 방법을 통해 영상에 나오는 사람이 다음에 어떠한 행동을 취하는지를 예측하였고 흡연을 하는 동작, 사물을 집는 동작 등에 대하여 인간의 행동에 가까운 모습으로 시각화하였다. Sitová 등 (2016)은 스마트폰에서 hand movement, ortientation and grasp (HMOG) 생체 정보의 특징에 관련된 멀티 모달(mutimodal) 데이터를 이용하여 연속적인 인증에 활용할 수 있는 방법을 제안하였다. 여기서 멀티 모달은 다양한 감각 기관을 통해 관찰되는 특징으로, 본 연구에서는 멀티 뷰의 하위 개념으로 사용하였다 (Lahat 등, 2015). Taylor 등 (2010)은 CNN과 restricted Boltzmann machine (RBM)을 응용한 convolutional gated RBM (convGRBM) 모형을 통하여 국소 공간 정보를 학습하고 모든 공간 정보가 공유하는 가중치를 학습시키는 방법을 제시하였다. 이를 통해 영상에 나오는 인간의 행동이 손뼉을 치는 동작인지, 뛰는 동작인지 여러 가지 동작이 어떤 행동인지 분류하는 문제에서 높은 성공률을 보인 바 있다.

2.2. 의학 분야

의학 분야에서는 의료 장비에 대한 기술의 발전에 따라 의료 영상 기술이 발전하여 초음파, MRI, CT, fMRI 등의 이미지 데이터가 과거에 비해 훨씬 정교해졌다. 하지만 여전히 의학적 진단에 대해서는 인간이 직접 진단을 해야 하고 때로는 전문가의 눈으로 진단하기 어려운 문제가 있는 경우도 있다. 때문에 의학 분야에서는 이미지 분할(segmentation), 컴퓨터 지원 진단(computer-aided diagnosis), 컴퓨터 지원 탐지(computer-aided detection) 등의 분야에서 이러한 문제를 해결하려는 연구가 이루어지고 있다 (Shen 등, 2017). 이때 멀티 뷰 기법을 통해 데이터의 특징을 학습하는 것이 도움이 될 수 있다. 예를 들어, 여러 종류의 의료 장비에서 얻은 신경 이미지(neuroimaging) 데이터를 멀티 뷰 기법을 이용하면 뇌의 구조를 더욱 잘 이해할 수 있다 (Uludağ와 Roebroeck, 2014). 이미지 분할에서는 뇌 영상의 경우 뇌의 사진이 아닌 두개골의 사진과 같은 것을 분리해 내는 것이 중요하다. Kiros 등 (2014)은 기존의 특징 기반 이미지 분할 방법과는 다르게 멀티 뷰 데이터의 여러 개의 스케일과 깊이로부터 특징을 학습하는 이미지 분할 방법을 제시하였다. 컴퓨터 지원 진단에서 Liu 등 (2015b)은 신경 이미지 데이터를 멀티 뷰 기법을 적용한 통합을 하여 알츠하이머를 가진 뇌를 분류함으로써 높은 적중률을 보인 바 있다. 진단 분야에서 Pohl 등 (2014)은 잠재적으로 치명적인 관상 동맥 질환을 진단하기 위하여 MRI, OCT, CT 등의 이미지 데이터를 융합하는 방법을 제안한 바 있다.

의학 분야에서의 문제는 개인 정보 보호법에 의해 다양한 의학 데이터를 수집하기가 어려워 소수의 데이터만이 학습이 가능한 경우가 많다는 것이다. 이러한 한계점을 극복하기 위하여 다른 데이터 도메인에서 이미 학습한 사전 지식을 바탕으로 학습을 하는 전이 학습(transfer learning) 방법과 이미지의 회전, 이동, 뒤집기 등을 이용한 데이터 확장(data augmentation) 방법을 통하여 이 문제를 해결하는 연구를 하였다 (Yu 등, 2017).

2.3. 정보 검색 분야

정보 검색 분야에서는 텍스트와 이미지, 오디오와 비디오 등 여러 가지 종류의 데이터에 대하여 한 가지 뷰의 데이터를 사용하여 연관되는 다른 종류의 뷰의 데이터를 찾는 문제에 멀티 뷰 기법을 적용하는 연구가 이루어져 왔다. 이를 통해 관련된 태그와 대응되는 이미지가 무엇인지 혹은 반대로 이미지를 통해 대응되는 문장을 검색하는 할 수 있는 시스템을 구현할 수 있고, 영화 이미지를 검색했을 때 관련된 리뷰를 찾아주는 추천 시스템에도 응용될 수 있다. Bai 등 (2010)은 latent semantic indexing (LSI)를 지

도학습기법적 방법으로 응용한 supervised semantic indexing (SSI) 모형을 제시하여 텍스트 데이터 쿼리가 주어졌을 때 관련된 문서의 순위를 파악할 수 있는 방법을 연구하였다. Wang 등 (2014)은 딥 러닝 기법인 Stacked Autoencoder를 응용하여 이미지 데이터를 통해 관련된 텍스트 데이터의 정보를 찾아낼 수 있는 방법을 연구하였다.

정보 검색 분야에서의 어려움은 의미론적인 차이를 고려해야 하는 문제와 다른 뷰의 데이터 간의 정보를 융합해야 한다는 문제에 있다 (Bokhari와 Hasan, 2013). Frome 등 (2013)은 텍스트 데이터를 통해 레이블 간의 의미론적인 관계를 학습하고 이미지 데이터를 의미론적인 임베딩 공간으로 매핑할 수 있는 deep visual-semantic embedding (DeViSE)을 통해 멀티 뷰 데이터의 의미론적 차이를 고려하는 연구를 하였다. Ngiam 등 (2011)은 오디오와 비디오 등 다른 뷰의 데이터를 서로가 공유하는 은닉층을 통해 학습시켜 두 데이터 간의 정보를 융합하는 방법을 제안하였다.

2.4. 표정 인식 분야

표정 인식 분야에서는 인간의 표정에서 나타나는 여러 가지 감정을 컴퓨터가 자동으로 인식하도록 하는 것이 이슈였다. 인간의 표정 변화는 주름, 팽창과 같은 사소한 변형과 눈썹, 입, 코 등과 같은 주요한 부위의 변형으로 표현되는 복잡한 구조를 띄어 컴퓨터가 인식하기에 쉽지 않다 (Kumari 등, 2015). 기존의 방법에는 얼굴의 이미지의 질감에 대한 정보를 픽셀 단위로 표현한 후 얼굴 표현의 특징을 학습하는 방법이 있다. 하지만 이 방법은 조명의 변화나 차폐의 변화에 매우 민감하다는 문제점이 있다 (Jeni 등, 2013). 또 다른 방법으로는 얼굴의 주요한 부분의 움직임의 표현을 학습하는 방법이 있다. 하지만 표정의 미묘한 변화를 효과적으로 포착할 수 없기 때문에 이러한 표현을 구별할 수 없다는 문제가 있다 (Lorincz 등, 2013).

이러한 문제를 해결하기 위하여 여러 가지 멀티 뷰 기법이 연구되고 있다. Zhang 등 (2015)은 이미지 질감에 대한 정보와 움직임의 표현에 대한 정보를 통합하는 연구를 수행하였다. Ding과 Tao (2015)는 사진의 부분마다 여러 개의 CNN을 이용하여 특징을 추출한 후 Autoencoder를 이용하여 특징을 통합하는 방식의 멀티 뷰 기법으로 사진의 조명, 포즈, 표정의 변화에 로버스트한 모형을 연구하였다. Jaques 등 (2017)은 얼굴 사진의 일부분이 소실됐을 때 multimodal autoencoder (MMAE) 모형을 통해 감정을 예측하여 소실된 부분을 복원하는 연구를 하였다.

3. 멀티 뷰 학습 기법의 분류

멀티 뷰 기법의 분류에 앞서 멀티 뷰 학습이 성공하기 위한 두 가지 학습 원리를 살펴보겠다 (Xu, 2013). 첫 번째는 일치의 원리(consistency principle)로 다른 뷰 사이의 일치 여부를 최대화시켜야 한다. 두 개의 독립적인 $f^{(1)}, f^{(2)}$을 뷰 1과 뷰 2의 데이터를 학습할 수 있는 각각의 가설이라고 하자. 예를 들어, $f^{(1)}, f^{(2)}$은 분류 문제에서는 데이터의 레이블을 예측할 수 있는 함수이거나 뷰 1과 뷰 2의 데이터의 표현을 학습할 수 있는 함수가 될 수 있다. 동의의 원리(consensus principle)에서는 이 독립적인 두 개의 가설의 의견이 일치하길 바라는 원리로 이는 두 개의 의견이 불일치하는 경우를 최소화하는 문제이기도 하다. 동의의 원리는 식 (3.1)과 같이 표현된다.

$$\min \left(\left|\left| f^{(1)} - f^{(2)} \right|\right|_2^2 \right). \tag{3.1}$$

동의의 원리에 가장 대표적인 경우가 정준상관분석(canonical correlation analysis; CCA)이나 뷰 간에 매칭에 이용되는 방법들이다. CCA의 경우 두 뷰의 상관관계를 최대화할 수 있는 투영을 학습하는 식으

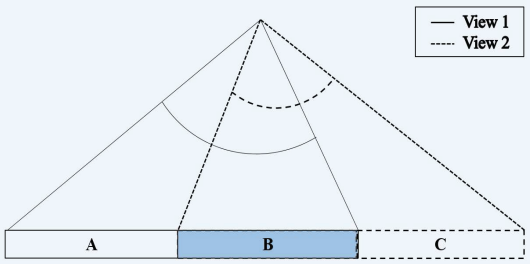

Figure 3.1. Illustration of consistency and complementary principles (Liu *et al.*, 2015a).

로 뷰 간의 일치성을 만족시킨다. 매칭에 이용되는 모형의 경우 이미지와 텍스트를 매칭하는 경우를 예로 들면, 이미지의 특징과 텍스트의 특징을 최대한 일치시켜 서로의 관계를 학습하는 방식으로 일치성을 만족시킨다.

두 번째는 보완의 원리(complementary principle)로 각 뷰의 정보가 서로를 보완할 수 있어야 한다. 각 뷰의 데이터는 다른 데이터가 갖고 있지 않은 정보를 갖고 있어야 학습의 성능을 높일 수 있다. 멀티 뷰 알고리즘은 이 두 가지 원리를 충족시키는 방향으로 발전되어 왔다. 대표적인 방법은 코트레이닝으로 뷰에 대한 독립적인 분류기가 서로를 보완하는 방식으로 학습한다.

Figure 3.1은 동의의 원리와 보완의 원리에 대한 그림 예시로 두 가지 뷰를 갖는 학습 대상이 잠재 공간(latent space)로 매핑된 것을 표현한 것이다. 잠재 공간으로 매핑되었을 때 A와 B는 뷰 1이 갖는 정보, B와 C는 뷰 2가 갖는 정보이다. 이때 뷰 1에서의 A 부분과 뷰 2에서의 C 부분은 두 뷰의 보완성을 의미하고 뷰 1과 뷰2에 공통적으로 존재하는 B는 두 뷰의 동의성을 의미한다. 즉, 두 뷰간에는 일치해야 하는 부분도 있고 서로 보완할 수 있는 부분도 있다는 것으로 멀티 뷰 학습은 이 두 가지 원리를 고려하는 것이 중요하다 (Liu 등, 2015a).

멀티 뷰 기법은 데이터의 통합 방식에 따라 크게 세 가지의 형태로 분류할 수 있다 (Ramchandram과 Taylor, 2017). 첫 번째는 데이터 차원의 통합으로 다른 뷰를 가진 데이터를 임베딩 방법을 통하여 통합하는 것이다. 데이터 차원의 통합의 대표적인 방법은 다변량 통계학의 고전적인 방법 중 하나인 CCA로 두 뷰의 선형적 연관성을 최대화하는 저차원의 부분 공간을 찾는 방식으로 뷰를 통합하는 방법이다. Andrew 등 (2013)은 CCA 기법에 딥 러닝 기법을 접목하여 기존의 CCA나 kernel CCA (KCCA)보다 뷰 간의 상관관계를 더욱 높이는데 성공하였다.

두 번째는 분류기 차원에서의 통합으로 뷰에 따라 각각 학습시킨 분류기들의 예측을 통합하는 방식이다. 대표적인 방법은 코트레이닝과 앙상블(ensemble)으로 각 뷰의 데이터를 통해 학습된 분류기끼리 서로 학습을 도와주는 방법이다. Blum과 Mitchell (1998)은 웹페이지가 강의에 관련된 페이지인지를 분류하는 연구를 진행하였는데 이때 코트레이닝 기법을 사용하였다. 강의 웹페이지 분류 문제에서 두 개의 뷰를 사용하여 레이블이 있는 소수의 데이터를 통해 레이블이 없는 다수의 데이터의 레이블을 생성하는 실험 결과 단일 뷰를 사용할 때보다 두 가지의 뷰를 사용할 때 더 좋은 성과를 거두었다.

세 번째 방법은 데이터 뷰마다 적절한 신경망을 통하여 표현을 학습하여 신경망 은닉층에서 통합하는 것이다. Srivastava와 Salakhutdinov (2012)은 멀티 모달 deep Boltzmann machine (DBM) 모형을 통해 이미지와 텍스트, 오디오와 비디오 같은 다른 형태를 갖는 데이터를 학습한 결과, 분류 문제에서 단일 뷰를 사용하는 딥 러닝 모형이나 support vector machine (SVM) 기반 모형을 사용하였을 때보다 뛰어난 성과를 거두었다. Ngiam 등 (2011)은 비디오만으로 이루어진 단일 모달(modality)보다 오디오와 비디오와 같은 멀티 모달에서 교차 모달 특징을 추출해 낼 수 있는 방법을 연구하였다. 이때 멀티 모달

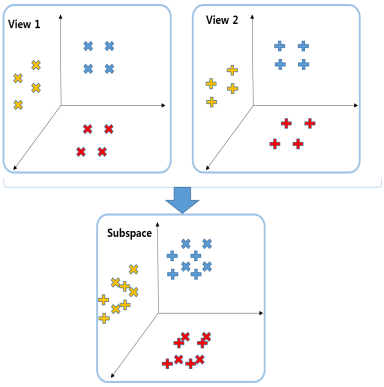

Figure 3.2. Data integration: projection method from each view onto common feature space.

융합, 교차 모달 학습, 공유 표현 학습 세 단계로 나누어 연구를 진행하였다. 본 장에서는 각각의 통합 방식에 대해서 어떠한 방식으로 멀티 뷰 데이터를 통합하는지 구체적으로 살펴보겠다.

3.1. 데이터 차원의 통합

다른 뷰를 가진 데이터는 뷰마다 다른 성질을 갖게 된다. 예를 들어, 텍스트 데이터는 이산적인 형태로 데이터가 존재하는 경우가 많고 희박한(sparse) 형태로 존재하는 경우가 많다. 반면에 이미지 데이터의 경우 픽셀 데이터가 연속형에 가까운 형태로 존재한다. Figure 3.2는 데이터 차원의 통합을 표현한 것으로 뷰 1과 뷰 2에 대하여 두 개의 뷰가 공유하는 하나의 특징 공간으로 투영하는 식으로 데이터의 특징을 통합하는 것으로 대표적으로 CCA가 있다.

CCA는 두 다차원 변수 간의 선형 관계를 상호 연관시키는 방법이다. 두 가지 변수 집합에 대한 기저 벡터(basis vector)를 찾는 문제로 기저 벡터에 대한 변수의 투영 간의 상관관계를 서로 극대화하는 문제로 볼 수 있다 (Hardoon 등, 2004). CCA는 다른 뷰를 가진 데이터 간의 상호 연관성을 최대화하는 선형 결합을 찾아냄으로써 데이터 간에 공유하는 부분 공간을 찾을 수 있다. $\mathbf{X}^{(1)} \in \mathbf{R}^{d_1 \times n_1}$는 뷰 1에 대한 데이터 행렬, $\mathbf{X}^{(2)} \in \mathbf{R}^{d_2 \times n_2}$는 뷰 2에 대한 데이터 행렬, $\mathbf{w}_1$는 $\mathbf{X}^{(1)}$를 선형 결합하는 가중치 벡터, $\mathbf{w}_2$는 $\mathbf{X}^{(2)}$를 선형 결합하는 가중치 벡터일 때, CCA는 식 (3.2)와 같이 표현된다 (Hotelling, 1936):

$$(\mathbf{w}_1^*, \mathbf{w}_2^*) = \arg\max_{\mathbf{w}_1, \mathbf{w}_2} \operatorname{corr}\left(\mathbf{w}_1^T \mathbf{X}^{(1)}, \mathbf{w}_2^T \mathbf{X}^{(2)}\right) = \arg\max_{\mathbf{w}_1, \mathbf{w}_2} \mathbf{w}_1^T C_{12} \mathbf{w}_2 \tag{3.2}$$
$$\text{subject to } \mathbf{w}_1^T C_{11} \mathbf{w}_1 = 1,\ \mathbf{w}_2^T C_{22} \mathbf{w}_2 = 1$$

여기서, C_{11}, C_{12}, C_{22}은 공분산 행렬이다. 결국 CCA는 $\mathbf{X}^{(1)}$의 선형 결합과 $\mathbf{X}^{(2)}$의 선형 결합의 상관관계를 최대화하는 가중치 벡터 $(\mathbf{w}_1^*, \mathbf{w}_2^*)$를 찾는 문제로 볼 수 있다.

CCA는 데이터의 선형성을 고려할 수 있는 모형으로 비선형성을 갖고 있는 데이터의 특징을 추출해 내는데 적합하지 않다. 이를 보완한 KCCA는 고차원의 특징 공간으로 데이터를 투영하여 데이터의 비선형성을 표현할 수 있다 (Crisitianini 등, 2002). 예를 들어 이미지 데이터의 경우 데이터에 선형성보다는 비선형성이 나타날 때가 많아 KCCA 방법을 통한 멀티 뷰 데이터 통합이 유용하다.

KCCA는 두 뷰의 데이터를 비선형 투영을 하는 쌍을 찾는 문제로 뷰 1의 데이터 행렬 $\mathbf{X}^{(1)} = (\mathbf{x}_1^{(1)}, \mathbf{x}_2^{(1)}, \ldots, \mathbf{x}_{n_1}^{(1)})$, $X^{(1)} \in \mathbf{R}^{d_1 \times n_1}$과 뷰 2의 데이터 행렬 $\mathbf{X}^{(2)} = (\mathbf{x}_1^{(2)}, \mathbf{x}_2^{(2)}, \ldots, \mathbf{x}_{n_2}^{(2)})$, $X^{(2)} \in \mathbf{R}^{d_2 \times n_2}$를 고차원의 특징 공간인 재생 커널 힐버트 공간(reproducing kernel Hilbert space; RKHS)

H_1, H_2로 매핑한다. 이에 대해 뷰1의 입력 공간에서 정의 되는 $\mathbf{x}_i^{(1)} \in X_1$의 H_1으로의 매핑은 $\Phi_1 : X_1 \to H_1$로 표현할 수 있고 뷰 2의 $\mathbf{x}_i^{(2)} \in X_2$에 관한 매핑 $\Phi_2 : X_2 \to H_2$ 역시 마찬가지이다. 따라서 KCCA는 $(f_1(\mathbf{X}^{(1)}), f_2(\mathbf{X}^{(2)}))$의 상관관계를 극대화하는 매핑 함수 $f_1^* \in H_1$, $f_2^* \in H_2$를 찾는 것을 목표로 하며 이는 식 (3.3)과 같이 정의 될 수 있다 (Cai 등, 2016):

$$(f_1^*, f_2^*) = \arg\max_{f_1, f_2} \operatorname{corr}\left(f_1\left(\mathbf{X}^{(1)}\right), f_2\left(\mathbf{X}^{(2)}\right)\right). \tag{3.3}$$

수식 (3.3)을 풀기위한 커널 트릭을 살펴보면 먼저 뷰 1에 관한 커널 함수 k_1은 $k_1 : X_1 \times X_1 \to \mathbf{R}$이고 모든 $\mathbf{x}_i^{(1)}, \mathbf{x}_j^{(1)} \in X_1$에 대하여 $k_1(\mathbf{x}_i^{(1)}, \mathbf{x}_j^{(1)}) =< \Phi_1(\mathbf{x}_i^{(1)}), \Phi_1(\mathbf{x}_j^{(1)}) >$로 정의 된다. 여기에서, $<$, $>$은 H_1에서 정의된 내적이다. 뷰 2에 대한 k_2 역시 마찬가지로 정의된다. $f_1(\mathbf{X}^{(1)}), f_2(\mathbf{X}^{(2)})$은 데이터의 커널의 선형 결합으로 표현될 수 있고 각각의 요소들은 식 (3.4)와 같다.

$$f_1\left(\mathbf{X}^{(1)}\right) = \alpha_1^T k_1\left(\cdot, \mathbf{x}_j^{(1)}\right), \qquad f_2\left(\mathbf{X}^{(2)}\right) = \alpha_2^T k_2\left(\cdot, \mathbf{x}_j^{(2)}\right), \tag{3.4}$$

여기서 α_1과 α_2는 데이터의 커널을 선형 결합으로 연결해주는 가중치 벡터이며 $k_1(\cdot, \mathbf{x}_j^{(1)})$은 i번째 요소가 $k_1(\mathbf{x}_i^{(1)}, \mathbf{x}_j^{(1)})$인 모든 i에 대한 표현이다.

이제 CCA의 식 (3.2)의 $\mathbf{w}_1$, $\mathbf{w}_2$를 대신하여 $f_1(\mathbf{X}^{(1)}), f_2(\mathbf{X}^{(2)})$를 넣고 전개를 하면 식 (3.5)를 도출할 수 있게 되고 KCCA는 결국 $(f_1(\mathbf{X}^{(1)}), f_2(\mathbf{X}^{(2)}))$의 상관관계를 극대화하는 (α_1^*, α_2^*)를 찾는 문제가 된다.

$$(\alpha_1^*, \alpha_2^*) = \arg\max_{\alpha_1, \alpha_2} \alpha_1^T \mathbf{K}_1 \mathbf{K}_2 \alpha_2 \quad \text{subject to } \alpha_1^T \mathbf{K}_1^2 \alpha_1, \alpha_2^T \mathbf{K}_2^2 \alpha_2, \tag{3.5}$$

여기서 $\mathbf{K}_1$는 중심화된 그램(gram) 행렬로 $\mathbf{K}_1 = \mathbf{K} - \mathbf{K1} - \mathbf{1K} + \mathbf{1K1}$이고, $\mathbf{K}$는 (i,j)번째 요소가 $k_1(x_{1i}, x_j)$인 행렬이고 $\mathbf{1}$은 모든 요소가 1인 행렬이다. $\mathbf{K}_2$ 역시 $\mathbf{K}$의 (i,j)번째 요소가 $k_2(x_{1i}, x_j)$인 행렬에 대해 마찬가지로 정의 된다.

CCA는 두 개 이상의 뷰에 대해서 사용할 수 있는 generalized CCA (GCCA)로 확장될 수 있다 (Horst, 1961). GCCA 모형은 J개의 다른 뷰가 있을 때 공유된 표현인 $\mathbf{G} \in \mathbf{R}^{r \times N}$를 찾는 문제로서 식 (3.6)과 같이 표현할 수 있다:

$$\arg\min_{\mathbf{W}_j, \mathbf{G}} \sum_{j=1}^{J} \left|\left|\mathbf{G} - \mathbf{W}_j^T \mathbf{X}^{(j)}\right|\right|_F^2 \quad \text{subject to } \mathbf{G}\mathbf{G}^T = \mathbf{I}_r, \tag{3.6}$$

여기서 $\mathbf{X}^{(j)} \in \mathbf{R}^{d_j \times N}$는 j번째 뷰의 데이터, N_j와 d_j은 j번째 뷰의 데이터의 수와 차원, r은 학습된 표현의 차원, $\mathbf{W}_j \in \mathbf{R}^{d_j \times r}$는 j번째 뷰의 선형 변환을 위한 가중치 행렬, $||\ ||_F$는 프로베니우스 노름(Frobenius norm)이다. CCA가 각 뷰에 대해서의 투영을 학습하지만 GCCA를 이용하여 뷰와 독립적인 공유된 표현을 학습할 수 있다.

이외에도 멀티 뷰 학습에서 CCA 기법은 여러 발전 과정을 거쳐 왔다. Farquhar 등 (2006)은 KCCA를 이용하여 두 개의 다른 SVM을 하나의 최적화 문제로 해결한 SVM-2K를 제시하였다. 딥 러닝 방법을 적용한 CCA도 연구되었는데 이는 5.1장에서 소개하겠다.

3.2. 분류기 차원의 통합

분류기 사이의 결과를 통합하는 방법의 대표적인 방법으로는 코트레이닝이 있다. 코트레이닝은 1998년에 Blum과 Mitchell에 의해 처음 제시되었는데 레이블이 일부만 있을 때 레이블을 생성해서 학습하

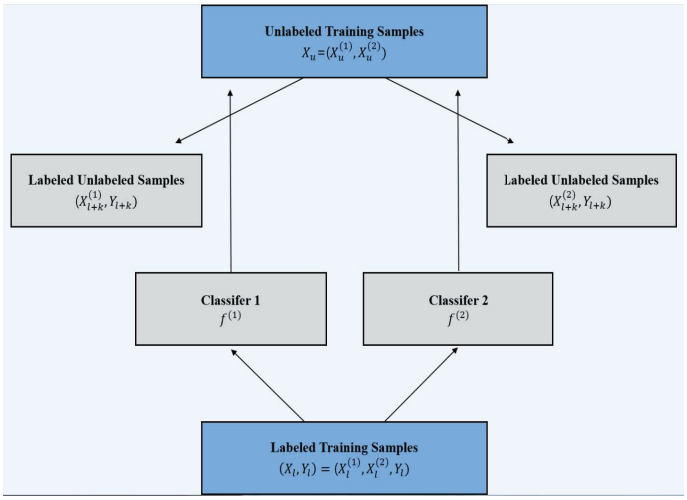

Figure 3.3. Flow chart of co-training learning.

는 준지도학습 기법에서 쓰이는 방법이다. 코트레이닝의 학습 방법을 살펴보면 먼저 두 개의 뷰에 대하여 두 개의 분류기를 각각 학습시킨다. 코트레이닝 알고리즘을 적용하기 위해서는 다음 세 가지 가정이 성립해야 한다 (Zhu, 2006): (i) 데이터 인스턴스(instance) $\mathbf{X}$ 벡터는 $\mathbf{X} = (\mathbf{X}^{(1)}, \mathbf{X}^{(2)})$으로 특징을 분리할 수 있어야 한다. (ii) $\mathbf{X}^{(1)}$ 혹은 $\mathbf{X}^{(2)}$ 각 뷰만으로 분류 예측이 가능해야 한다. (iii) $\mathbf{X}^{(1)}$와 $\mathbf{X}^{(2)}$는 주어진 레이블 안에 조건부 독립이어야 한다. 여기서 각 뷰의 데이터 인스턴스는 $x^{(1)} \in \mathbf{X}^{(1)}$, $x^{(2)} \in \mathbf{X}^{(2)}$이다.

Figure 3.3은 코트레이닝의 학습 과정을 나타낸 그림이다. 학습 데이터에서 데이터 인스턴스는 n개가 존재하지만 레이블 $\mathbf{Y}$는 l개만이 존재하여 $(\mathbf{X}_l^{(1)}, \mathbf{X}_l^{(2)}) = \{(x_{1:l}, y_{1:l})\}$이 처음에 학습 가능한 데이터이고 나머지 레이블이 없는 u개의 학습 데이터는 $\mathbf{X} = \{x_{l+1:n}\}$로 표현된다. 먼저 가정 (i)에 따라 레이블이 있는 학습 데이터 $\mathbf{X}_l$는 $\mathbf{X}_l = (\mathbf{X}_l^{(1)}, \mathbf{X}_l^{(2)})$로 나누어져 $(\mathbf{X}_l, \mathbf{Y}_l) = (\mathbf{X}_l^{(1)}, \mathbf{X}_l^{(2)}, \mathbf{Y}_l)$이 된다. 분류기 $f^{(1)}$은 레이블이 있는 뷰1의 데이터 $(\mathbf{X}_l^{(1)}, \mathbf{Y}_l)$로부터 학습을 하여 $\mathbf{X}_u$의 레이블을 예측한다. 마찬가지로 분류기 $f^{(1)}$은 레이블이 있는 뷰2 데이터 $(\mathbf{X}_l^{(2)}, \mathbf{Y}_l)$로부터 학습을 하여 $\mathbf{X}_u$의 레이블을 예측한다. $f^{(1)}$은 예측한 레이블 중 가장 신뢰할 수 있는 k개의 $f^{(1)}(x^{(1)})$를 $f^{(1)}$의 레이블로 가르쳐 주고, 역시 가장 신뢰할 수 있는 k개의 $f^{(2)}(x^{(2)})$를 $f^{(1)}$의 레이블로 가르쳐 준다 (Zhu와 Goldberg, 2007). 이후에 원래의 훈련 데이터에 있던 l개의 레이블과 생성된 k개의 레이블이 있는 데이터를 통하여 테스트 데이터의 레이블을 예측할 수 있다.

코트레이닝의 세 가지 가정을 살펴보면 멀티 뷰 학습의 학습 원리를 따른 다는 것을 알 수 있다. 코트레이닝에서는 약간 완화된 일치의 원리인 동의의 원리를 사용한다. 분류기 $f^{(1)}, f^{(2)}$ 사이의 예측이 너무 같으면 서로 보완이 되지 않는다. 예를 들어 문서를 분류할 때 $f^{(1)}, f^{(2)}$가 계속 같은 문서라고 분류하면 서로 보완성이 떨어질 것이다. 하지만 (ii)의 가정에서 독립적인 분류기가 각각 분류 예측이 가능하다고 했기 때문에 어느 정도는 같은 예측을 해야한다. 예를 들어, 문서 분류에 있어서 $f^{(1)}, f^{(2)}$가 어느 정도는 같은 문서라고 예측하는 것을 허용해야 한다. 따라서 이를 어느 정도 허용하는 Dasgupta 등 (2002)이 유도한 동의의 원리에 대한 제약식은 식 (3.7)과 같다.

$$P\left(f^{(1)} \neq f^{(2)}\right) \geq \max\left(P_{\text{err}}\left(f^{(1)}\right), P_{\text{err}}\left(f^{(2)}\right)\right). \tag{3.7}$$

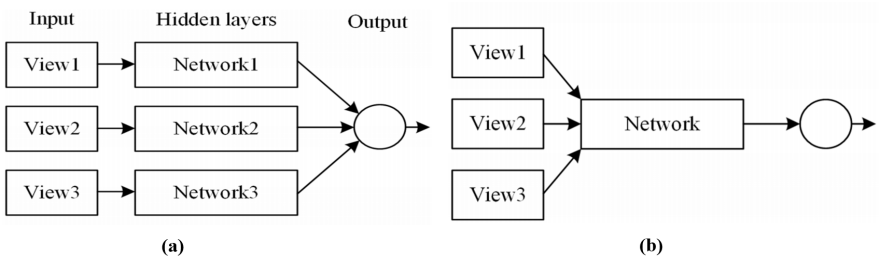

Figure 3.4. Two kinds of the multi-view strategy: (a) One-view-one-network strategy, (b) Multi-view-one-network strategy (Kang *et al.*, 2017).

$P_{\text{err}}(f^{(1)}), P_{\text{err}}(f^{(2)})$은 각각의 분류기가 잘못 예측할 확률이다. 즉, 식 (3.7)의 왼쪽 항은 두 분류기의 의견이 일치하지 않을 확률이고 오른쪽은 각 분류기의 틀릴 확률을 최대로 허용하는 정도이다. 왼쪽 항이 오른쪽 항의 제약으로써 두 분류기의 불일치 확률을 최소화하는 것은 각 분류기의 틀릴 확률을 최대로 허용 정도를 최소화하는 문제가 된다.

코트레이닝의 알고리즘의 핵심은 서로 보완적인 다른 두 개의 분류기가 서로를 학습시켜준다는 것으로 두 분류기의 예측이 서로 최대한 일치해야 좋은 성능을 발휘할 수 있다 (Du 등, 2011). 이는 멀티 뷰 학습의 보완의 원리와 상응한다. Sousa와 Gama (2017)는 코트레이닝 알고리즘에서 가정 (iii)의 뷰 간에 조건부 독립성이 만족할 경우 단일 뷰를 이용해서 학습하는 경우보다 더 좋은 성능을 보인다는 것을 실험적으로 밝혔다. 뷰 간에 조건부 독립성이 약할 경우 뷰의 특징들 간의 조건부 상호 정보가 없어지게 되어 보완성이 약해진다. 또한 생성된 레이블의 정보력이 떨어지게 되기 때문에 코트레이닝 알고리즘의 성능이 떨어지게 된다 (Nigam과 Ghani, 2000).

코트레이닝 기법은 여러 가지 방법으로 응용되어 지속적으로 발전해왔다. Nigam과 Ghani (2000)는 co-expectation maximization (Co-EM) 방법을 제시하였는데, 이 방법은 각 뷰에 대해 EM 알고리즘을 통해 레이블이 없는 데이터에 확률적으로 레이블을 생성하는 방법이다. 코트레이닝이 학습과정에서 한 번에 소수의 레이블을 생성해나가는 것과 달리 Co-EM은 $f^{(1)}$이 $(\mathbf{X}_l^{(1)}, \mathbf{Y}_l)$을 이용하여 모든 레이블이 없는 데이터의 레이블을 각 알고리즘적 반복마다 레이블의 확률을 재추정하여 $f^{(2)}$가 학습할 수 있도록 돕는 방법이다. Sindhwani 등 (2005)은 코트레이닝의 목적 함수에 정규화 항을 추가하여 두 분류기 사이에 동의하지 않는 경우에 대해 벌점을 부여하였다. Zhou와 Li 등 (2005)은 tri-training 방법을 통하여 세 개의 분류기를 이용하여 학습시키는 방법을 제시하였다.

Kang 등 (2017)은 멀티 뷰 기법을 두 가지 통합 방법으로 구현하였는데 한 뷰에 대해 하나의 CNN 모형을 적용하는 방법(one-view-one-network strategy)과 다른 하나는 멀티 뷰 데이터를 하나의 CNN(multi-view-one-network strategy)으로 통합하는 방법이다. 한 뷰에 대해 하나의 CNN 모형을 적용하는 방법을 통합하는 방법은 전형적인 앙상블 방법으로 학습 구조는 Figure 3.4(a)와 같다. 멀티 뷰 데이터를 하나의 CNN으로 통합하는 방법은 학습된 표현간의 통합 기법으로 4.1장에서 다루겠다. 이 방법을 활용하여 Dou 등 (2017)은 폐 결절을 분류 하는데 사용하였다. 이 방법에서는 폐 결절의 복잡한 해부학적 환경을 고려하기 위해 3D CNN을 사용하여 공간적 정보를 최대한 활용하려 하였다. 또한 폐 결절마다 다양한 변형과 크기, 모양 등 다양한 특성을 갖고 있지만 다양한 종류의 결절의 특징에 대한 정보를 모으기 어렵다는 점에서 폐 결절 CT에 3가지의 각기 다른 수용 영역(receptive field)를 적용한 CNN 모형을 사용하였다. 여기서 수용 영역이란 타깃으로 삼는 3D 샘플을 포함하는 범위이다. 각각 다른 수용 영역의 크기로 학습시킨 CNN의 분류 예측 결과는 세 모형의 분류기의 앙상블로 결합된다. 따라서 테스트 데이터의 폐 결절의 j번째 후보군 I_j을 예측할 때 분류 예측을 위한 확률은

식 (3.8)과 같은 세 분류기의 선형결합으로 표현할 수 있다.

$$P\left(\hat{Y}_j = c|I_j\right) = \sum_{\phi\in\{1,2,3\}} \gamma_\phi P_\phi\left(\hat{Y}_j = c|I_j;\theta_\phi\right), \tag{3.8}$$

여기서 $\hat{Y}_j$는 후보군 I_j의 예측된 레이블이고, θ_ϕ는 3개의 CNN 모형의 각각의 모수이고, γ_ϕ는 세 분류기를 결합하는 가중치이다.

Adetiba와 Olugbara (2015)는 앙상블 기법을 이용하여 폐 암 여부를 예측하기도 하였다. 이 때 artificial neural networks (ANN)과 SVM을 이용하여 앙상블을 하였고 ANN 앙상블의 경우 폐암의 선별 및 조기 발견에서 높은 성능을 보였다.

3.3. 학습된 표현 간의 통합

신경망 기법은 비선형 변환을 통해 데이터의 특징을 학습할 수 있는 방법으로 멀티 뷰 학습에서 통합 방법은 다음과 같다. 먼저 각 뷰의 표현을 학습하는데 뛰어난 성능을 가진 모형으로 뷰마다 다른 신경망으로 학습을 시킨다. 예를 들어, 이미지와 관련 텍스트로 이루어진 멀티 뷰 데이터의 표현을 학습할 때 이미지 표현 학습에서 큰 성과를 거둔 CNN을 사용하고 텍스트의 언어적 순차성을 고려할 수 있는 RNN을 사용하여 각 뷰의 표현을 학습시키는 식이다. 신경망 기법을 통해 학습된 표현은 각 뷰의 신경망이 동시에 공유하는 은닉층과 연결되어 결합되고 결합된 은닉층에 연결된 가중치를 찾는 방식으로 학습이 이루어진다 (Ramachandram과 Taylor, 2017).

4. 딥 러닝 기법을 응용한 멀티 뷰 학습 기법의 분류 1: 지도학습 기법

본 장에서는 딥 러닝 기법을 이용한 멀티 뷰 학습에서 학습된 표현 간의 통합을 하는 방법에 대해서 설명하겠다. 기계학습 방법론은 크게 지도학습과 비지도학습으로 나눌 수 있고 준지도학습은 그 중간에 위치한다고 할 수 있다. 지도학습은 주로 회귀 분석 혹은 분류에 사용 될 수 있는 방법으로 데이터 변수가 주어졌을 때 출력 변수의 조건부 분포 혹은 함수를 학습하는 것을 목표로 한다. SVM, 의사 결정 나무 등이 이에 속한다. 비지도 학습은 데이터 자체의 분포 혹은 데이터의 표현을 학습하는 방법으로 클러스터링, 주성분분석(principal component analysis; PCA) 등이 이에 속한다 (Dey, 2016).

본 장과 다음 5장에서는 딥 러닝 기법들을 지도학습과 비지도학습 중 많이 사용되는 학습 기법으로 분류하였고 이를 응용한 멀티 뷰 기법을 소개하였다. 최근 딥 러닝에서 지도 학습 기법으로 사용되던 모형이 비지도 학습으로 사용되기도 하고 반대의 경우도 많이 일어나고 있지만 모형의 근본적인 학습 원리에 대한 이해를 돕고자하였다.

4.1. CNN 기반 기법

CNN을 기반으로 한 멀티 뷰 기법으로 데이터의 형태가 다른 멀티 모달 데이터에 적용할 수 있는 모형과 하나의 데이터를 여러 가지 뷰로 나눈 멀티 뷰 데이터에 대하여 적용할 수 있는 모형이 있다. 먼저 멀티 모달 데이터에 적용할 수 있는 모형으로는 multimodal CNN (m-CNN)이 있는데, 이 모형은 신경망 기법 중 하나인 CNN을 멀티 모달 데이터에 적용한 방법으로 이미지와 연관된 텍스트의 관계를 학습할 수 있는 방법이다 (Ma 등, 2015). 이미지와 관련된 텍스트를 학습시키는 문제는 여러 가지 경우를 고려해야 하기 때문에 쉽지 않다. 예를 들어, Figure 4.1의 왼쪽의 이미지에 “젖은 강아지가 흰 공을 쫓는다”라는 문장을 매칭 시키려 할 때, “강아지”라는 단어와 강아지 이미지와의 관계, “젖은 강아지”와 강아지 이미지와의 관계, “공”과 공 이미지와의 관계, “흰 공”과 공 이미지와의 관계, “강아지, 공”과 개와

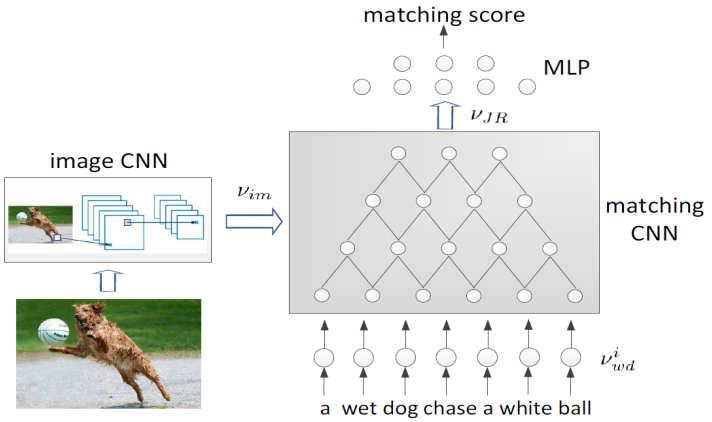

Figure 4.1. The word-level matching CNN (Ma *et al.*, 2015).

공의 이미지와의 관계 등 여러 가지 관계를 학습시켜야 되는 문제가 있다. 이때 해당 이미지에 연관된 단어들의 길이 또한 다르기 때문에 학습이 쉽지가 않다.

Figure 4.1은 m-CNN의 학습 구조로 m-CNN은 이미지 CNN, 매칭 CNN, multilayer perceptron (MLP)으로 이루어진 구조를 통해 이미지와 텍스트의 매칭 점수를 계산하여 이 문제를 해결하는 방법이다. 이미지 CNN을 통해 학습된 이미지 표현 v_{im}을 매칭 CNN에서 단어의 표현을 매칭한다. 이때 컨볼루션 유닛의 입력 값은 단어에 대한 벡터적 표현 v^i_{wd}와 k_{rp}개의 인접한 단어 벡터들이 이미지 표현 v_{im}과 함께 입력되며 이후의 층에서는 이전의 층에서 학습된 표현이 맥스 풀링(max pooling)을 거쳐 입력된다. 이는 수식 (4.1)과 같다.

$$\mathbf{v} = v^i_{wd} \left|\left| v^{i+1}_{wd} \right|\right| \cdots \left|\left| v^{i+k_{rp}-1}_{wd} \right|\right| v_{im}, \tag{4.1}$$

여기서 ||는 근처 k_{rp}개의 인근의 단어 벡터들을 긴 단어 벡터로 합치는 것을 의미하고 v^i_{wd}는 문장에서 i번째 단어(word)의 표현을 뜻한다. 이렇게 학습된 표현은 최종 층에서 텍스트와 이미지의 결합표현 v_{JR}을 생성할 수 있게 된다. 이후 MLP 과정에서는 결합 표현 s_{match}을 통해 스코어 함수 의 매칭 점수를 계산할 수 있게 되는데 이는 수식 (4.2)와 같다.

$$s_{\text{match}} = w_s(\sigma(w_h)(v_{JR}) + b_h) + b_s, \tag{4.2}$$

여기서 w_h와 b_h은 은닉층에서의 가중치와 편향이고, w_s와 b_s는 이미지와 텍스트의 매칭 점수를 계산하기 위한 가중치와 편향이다. σ은 비선형 활성 함수이다. m-CNN을 학습시키기 위한 목적 함수 $e_\theta(x_n, y_n, y_m)$는 식 (4.3)과 같다.

$$e_\theta(x_n, y_n, y_m) = \max(0, \mu - s_{\text{match}}(x_n, y_n) + s_{\text{match}}(x_n, y_m)), \tag{4.3}$$

여기서 $\theta = \{w_{im}, b_{im}, w_h, b_h, w_s, b_s\}$는 m-CNN 모형의 파라미터, (x_n, y_n)는 연관된 텍스트와 이미지의 쌍, (x_n, y_m)는 임의로 샘플링 된 연관 없는 이미지와 텍스트의 쌍을 의미한다. 목적 함수를 보면 알 수 있듯이 m-CNN 모형은 (x_n, y_n)의 매칭 점수가 (x_n, y_m)의 매칭 점수보다 마진 μ만큼 높게 하는 방향으로 학습된다.

앞서 3.2장에서 소개했던 Kang 등 (2017)이 제안한 멀티 뷰를 하나의 CNN에서 통합하는 방법의 학습 구조는 Figure 3.4(b)와 같다. 이 방법은 뷰마다 각각의 특징을 추출 후 특징을 통합하고 통합된 특징

을 CNN을 이용하여 학습하는 방법이다. 이 방법을 활용하여 Su 등 (2015)은 3D 사물을 인식하는 문제에서 2D 표현에 대하여 멀티 뷰 CNN을 사용하여 학습하는 방법을 제안하였다. 의자의 3D 이미지를 분류 예측하는 문제에서 멀티 뷰 CNN을 학습시키기 위해 먼저 의자의 사진을 여러 가지 각도로 찍어 여러 가지 뷰로 의자 이미지를 표현하였다. 각 뷰에 대하여 CNN의 2D 필터를 적용하여 특징을 추출한 후 각 뷰의 특징은 풀링층에서 통합되어 통합된 CNN 구조에서 다시 학습된다. 이 방법은 3D 필터를 적용하여 단일 뷰로 모형을 구현한 경우보다 더 좋은 성능을 보였다. 그 이유는 2D 표현의 경우 3D 표현에 비해 사물의 깊이에 대한 정보가 없지만 같은 입력 크기로 더 높은 해상도의 데이터를 얻을 수 있기 때문이다. 예를 들어, 3D 형태의 $30 \times 30 \times 30$ 형태의 복셀(voxel) 이미지 표현은 2D 형태의 164×164 크기의 픽셀 데이터로 대응될 수 있어 더 높은 공간적 해상도를 얻을 수 있다.

멀티 뷰를 하나의 CNN에서 통합하는 방법은 이후 Qi 등 (2016)이 멀티 뷰 CNN을 3D 데이터에 대하여 적용할 수 있게 확장하였고 단일 뷰에 대한 CNN보다 분류 예측에 있어 더 좋은 성능을 보였다. Kang 등 (2017)은 의학 분야에서 폐에 대한 CT 이미지에서 3D 데이터에 대한 멀티 뷰 CNN을 사용하여 폐 결절을 분류 예측하기도 하였다. Tatulli와 Hueber (2017)는 음성 인식 문제에서 오디오 신호 데이터 없이 초음파 영상과 입모양의 사진만을 이용하였다. 이때 두 이미지에 대하여 각각의 다른 CNN을 통하여 특징을 추출하였고 이 두 특징을 통합하는 층을 만들어 두 특징을 통합하였다. 이 방법을 통하여 PCA를 통하여 특징을 추출하였을 때 보다 더 좋은 결과를 보였다.

4.2. RNN 기반 기법

RNN은 순열적인 데이터에 주로 사용되는 신경망 기법으로 시계열 데이터, 텍스트 데이터, 음성데이터의 학습에 주로 사용된다. Cho 등 (2014b)은 언어를 다른 언어로 번역할 수 있는 RNN 인코더-디코더(encoder-decoder) 모형을 제안하였다. 언어의 번역의 문제에서 같은 언어라도 언어가 다르면 문장의 길이가 달라진다. RNN 인코더-디코더 모형은 다른 언어 간의 문장의 길이가 달라도 학습할 수 있는 인코더-디코더 구조를 통해 이러한 문제를 해결한 모형이다.

RNN 인코더-디코더 모형을 살펴보면 두 개의 RNN 모형으로 구성되어 첫 번째 RNN 모형은 순열 데이터를 고정된 길이의 벡터 표현으로 인코딩하고 두 번째 RNN 모형은 다른 순열 데이터를 벡터 표현으로 디코딩한다. 이 모형의 특징은 인고딩 된 특징을 고정된 길이의 벡터 표현으로 압축을 하고 이 압축된 특징을 바탕으로 다른 언어의 데이터를 디코딩 하는 방식으로 다른 뷰의 데이터를 통합한다. 이 모형의 최종 목표는 길이가 T인 입력 언어 데이터 $\mathbf{x}_t = (x_1, \ldots, x_T)$가 주어졌을 때 길이가 $T^{'}$인 출력 언어 데이터 $\mathbf{y}_t = (y_1, \ldots, y_{T'})$의 조건부 분포 $p(\mathbf{y}_t|\mathbf{x}_t)$를 학습하는 것이다.

인코더 부분을 살펴보면 RNN으로 순열 데이터 $\mathbf{x}_t$를 학습하고 학습이 진행됨에 따라 t시점의 RNN의 은닉 상태(hidden state) $\mathbf{h}_t$는 이전 시점의 은닉 상태와 t시점의 데이터를 통해 식 (4.4)와 같이 업데이트된다.

$$\mathbf{h}_t = \sigma_1(\mathbf{h}_{t-1}, \mathbf{x}_t) \tag{4.4}$$

여기서 t는 시점에 대한 인덱스이고, σ_1는 비선형 활성 함수이다. 디코더 부분에서는 또 다른 RNN으로 은닉 상태 $\mathbf{h}_t$, $\mathbf{y}_{t-1}$, 요약 상태 $\mathbf{c}$가 주어졌을 때 $\mathbf{y}_t$를 예측할 수 있게 학습되며 여기서 요약 상태 $\mathbf{c}$는 인코더 RNN의 입력 데이터의 마지막 순열을 학습했을 때의 은닉 상태이다. 이 요약 상태 $\mathbf{c}$는 인코더의 입력 데이터의 특징을 압축하는 역할을 한다. 디코더 부분의 은닉 상태는 식 (4.5)와 같다.

$$\mathbf{h}_t = \sigma_2(\mathbf{h}_{t-1}, \mathbf{y}_{t-1}, \mathbf{c}) \tag{4.5}$$

여기서 σ_2는 비선형 활성 함수이다. 이를 통해 의 조건부 분포를 식 (4.6)과 같이 구할 수 있게 된다.

$$P(\mathbf{y}_t|\mathbf{y}_{t-1},\ldots,\mathbf{y}_1,\mathbf{c}) = \sigma_2(\mathbf{h}_t,\mathbf{y}_t,\mathbf{c}) \tag{4.6}$$

RNN 모형을 학습시키기 위한 목적 함수로 조건부 로그 가능도 함수가 사용되며 이는 식 (4.7)과 같다.

$$\max_{\theta} \frac{1}{N}\sum_{n=1}^{N} \log P_\theta(\mathbf{y_n}|\mathbf{x_n}), \tag{4.7}$$

여기서 θ는 RNN 인코더-디코더 모형의 모수, 훈련 데이터에서 $(\mathbf{x_n},\mathbf{y_n})$는 입력 언어와 출력 언어의 데이터 쌍이다. 목적 함수를 보면 조건부 로그 가능도 함수의 사용함으로서 한 언어가 입력되었을 때 그에 해당하는 다른 언어가 가장 높은 확률로 추출되는 방향으로 학습하는 것임을 알 수 있다.

이 모형은 한계점을 갖고 있지만 이를 개선하면서 발전해왔다. Cho 등 (2014a)은 RNN 인코더-디코더 모형이 입력되는 순열 데이터의 길이가 길수록 학습이 잘 안되는 점을 발견하였다. Sutskever 등 (2014)은 RNN 인코더-디코더 모형과 마찬가지로 입력되는 순열 데이터를 고정된 길이의 벡터 표현으로 매핑할 수 있는 seq2seq 모형을 제시하였다. 이때 long-short term memory (LSTM)을 이용하여 순열 데이터의 길이가 길어서 시간에 대한 종속성 심한 경우에도 학습이 가능하도록 하였다. Bahdanau 등 (2014)은 위 모형들이 고정된 길이의 벡터에 필요한 정보들을 압축하기 때문에 입력 데이터의 길이가 긴 경우 학습이 잘 안된다고 보고 입력 데이터의 길이에 따라 적응적으로 벡터 표현의 길이를 조절하는 방법을 제안하였다.

또 다른 RNN 기반 기법은 Mao 등 (2014)이 제시한 multimodal RNN (m-RNN)이다. 이 기법은 RNN 기법을 멀티 뷰 데이터에 대해 응용한 방법으로 이미지와 관련된 문장을 생성하거나 이미지와 관련된 문장을 검색할 수 있는 방법이다. Figure 4.2(a)는 RNN 모형의 학습 구조로, 단일 뷰의 데이터에 대해서는 사용되는 RNN 모형을 살펴보면 각 시간의 프레임마다 단어 층 $\mathbf{x}_t$, 순환(recurrent) 층 $\mathbf{r}_t$, 출력 층 $\mathbf{y}_t$로 구성되어있고 이는 식 (4.8)과 같다 (Elman, 1990).

$$\mathbf{x}_t = [\mathbf{w}_t\mathbf{r}_{t-1}], \qquad \mathbf{r}_t = \sigma_1(\mathbf{U}\cdot\mathbf{x}_t), \qquad \mathbf{y}_t = \sigma_2(\mathbf{V}\cdot\mathbf{r}_t), \tag{4.8}$$

여기서 $[\cdot]$는 벡터의 연결(concatenation)이며, $\mathbf{w}_t$는 t시점의 단어 벡터, $\mathbf{x}_t$는 $\mathbf{w}_t$와 $\mathbf{r}_{t-1}$를 연결한 벡터이다. 이 $\mathbf{x}_t$가 가중치 $\mathbf{U}$를 통해 연결되어 sigmoid 활성 함수 σ_1를 통해 t시점의 순환 층 $\mathbf{r}_t$에 입력된다. 마찬가지로 $\mathbf{r}_t$는 가중치 $\mathbf{V}$를 통해 연결되어 소프트맥스 활성 함수 σ_2를 통해 $\mathbf{y}_t$를 출력한다.

Figure 4.2(b)는 m-RNN의 학습 구조로, m-RNN은 전후 시점의 순환 층 $\mathbf{r}_t$가 서로 연결되어 시간적 특성을 고려할 수 있음을 알 수 있다. m-RNN은 언어 모형, 이미지 모형, 멀티 모달 모형의 세 가지로 구성된다. 언어 모형은 사전에 있는 단어의 특징 임베딩을 학습하여 순환 층에 언어의 시간적 맥락을 저장하는 단계이다. 처음에 입력층에는 원-핫 인코딩(one-hot encoding)된 단어가 입력층에 들어가서 희박한(sparse) 단어 벡터가 두 번의 임베딩을 통해 밀집한 단어 표현을 얻을 수 있다. RNN에서는 순환층에 단순히 단어 벡터와 전 시점의 순환층을 합친 벡터 형태로 입력을 했지만 m-RNN의 순환층에는 식 (4.9)와 같이 단어 벡터가 따로 더해지는 형태로 입력된다.

$$\mathbf{r}_t = \sigma_3(\mathbf{U}\cdot\mathbf{r}_{t-1} + \mathbf{w}_t). \tag{4.9}$$

이것은 $\mathbf{r}_{t-1}$를 $\mathbf{w}_t$와 같은 벡터 차원으로 매핑하기 위함이다. 여기서 σ_3는 rectified linear unit (ReLU) 활성 함수로 RNN에서 자주 나타나는 경사 감소 소멸(vanishing gradient) 문제를 방지하기 위하여 사용된다.

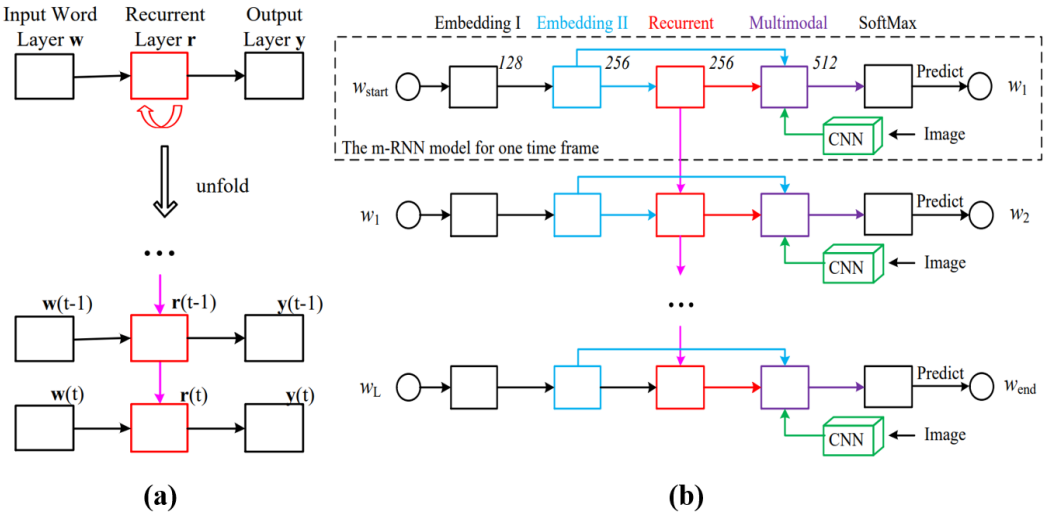

Figure 4.2. (a) The simple RNN model (b) The m-RNN model (Mao *et al.*, 2014).

다음으로 이미지 모형에서는 CNN 모형을 통하여 이미지의 특징을 추출하고 멀티 모달 모형에서 언어 모형에서 학습된 특징과 이미지 모형에서 학습된 특징이 결합된다. 학습된 특징을 결합하는 방식을 보면 다른 뷰의 특징을 벡터적 연결을 통해 통합했던 m-CNN과 달리 m-RNN에서는 멀티 모달 레이어에서 서로 다른 특징 정보들이 더해지는 식으로 결합되고 멀티 모달 레이어의 특징 벡터 $\mathbf{m}_t$은 식 (4.10)과 같다.

$$\mathbf{m}_t = \sigma_4(\mathbf{V}_w \cdot \mathbf{w}_t + \mathbf{V}_r \cdot \mathbf{r}_t + \mathbf{V}_I \cdot \mathbf{I}), \tag{4.10}$$

여기서 $\mathbf{I}$는 이미지 특징 벡터로 σ_4는 tanh 활성 함수이다. 단어 벡터와 순환 벡터, 이미지 특징 벡터가 각각 가중치로 연결되고 그 합으로 활성 함수에 입력되는 형태이다.

m-RNN을 학습시키기 위해서 목적 함수로 혼란도(perplexity)와 로그 가능도 함수를 결합된 형태를 사용한다. 혼란도는 데이터가 희박한 토픽 모형과 같은 언어 모형에서 사용되는 측도로 이미지 특징 $\mathbf{I}$가 주어졌을 때 단어의 순열 $w_{1:L}$의 혼란도는 식 (4.11)과 같이 계산된다.

$$\log_2 PP(\mathbf{w}_{1:L}|\mathbf{I}) = -\frac{1}{L}\sum_{n=1}^{L} \log_2 P(\mathbf{w}_n|\mathbf{w}_{1:n-1}, \mathbf{I}), \tag{4.11}$$

여기서 L은 단어의 수이다. 목적 함수는 i번째 문장의 문맥과 관련된 이미지가 주어졌을 때 단어들의 평균 가능도로 계산되며 이는 식 (4.12)와 같다.

$$C = \frac{1}{N}\sum_{n=1}^{N_s} L_i \log_2 PP\left(\mathbf{w}_{1:L_i}^{(i)}|\mathbf{I}^{(i)}\right) + \lambda_\theta ||\theta||_2^2, \tag{4.12}$$

여기서 N은 훈련 데이터에서의 단어의 수이고 N_s는 훈련 데이터에서의 문장의 수, L_i는 i번째 문장의 길이, λ_θ는 정규화 항, $\theta = \{\mathbf{U}, \mathbf{V}\}$는 m-RNN의 모수이다. m-RNN은 목적 함수 C를 최소화하는 방식으로 학습한다.

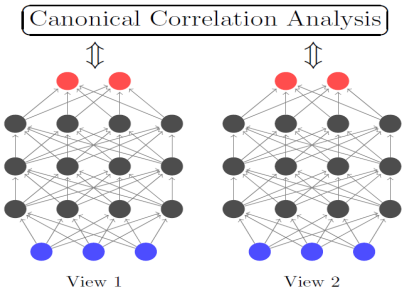

Figure 5.1. A schematic of DCCA (Andrew *et al.*, 2013).

5. 딥 러닝 기법을 응용한 멀티 뷰 학습 기법의 분류 2: 비지도학습 기법

5.1. CCA 기반 기법

멀티 뷰 데이터를 학습할 수 있는 비지도 학습 기법인 CCA는 선형적인 관계를 고려할 수 있지만 뷰 간의 비선형적인 관계를 학습할 수 없다는 문제점이 있다. KCCA는 비선형적인 표현을 학습할 수 있는 방법이지만 뷰 간의 심층적인 관계를 학습시킬 수 없다는 문제점이 있다. Deep CCA (DCCA)는 각 뷰마다 심층 비선형 매핑이 최대한 상관되도록 학습시키는 방법으로 뷰 사이의 심층적 관계를 딥 러닝 기법을 이용하여 학습시킬 수 있다 (Andrew 등, 2013). Figure 5.1은 DCCA의 학습 도안으로 학습 과정을 살펴보면 각각의 뷰에 대하여 신경망 모형을 학습시킨다. 식 (5.1)은 뷰 1을 학습시키는 수식으로 입력층에 첫 번째 뷰의 데이터 $x^{(1)}$이 학습되어 다음 층으로 들어가는 형식이다.

$$h_1 = \sigma\left(W_2^{(1)}x^{(1)} + b_1^{(1)}\right), \qquad f_1\left(x^{(1)}\right) = \sigma\left(W_l^{(1)}h_{l-1} + b_l^{(1)}\right), \tag{5.1}$$

여기서 h_1은 뷰 1에 대한 첫 번째 은닉층이고 h_{l-1}은 $(l-1)$번째 은닉층이고 $b_l^{(1)}$은 편향, $W_l^{(1)}$은 가중치 행렬이다. $f_1(x^{(1)})$은 출력층의 비선형 변환된 값으로 DCCA의 뷰 1에 대한 학습 방식은 전형적인 deep neural network (DNN)과 같고 뷰 2에 대해서도 같은 학습 구조를 통해 $f_2(x^{(2)})$를 구한다. 식 (5.2)는 DCCA의 목적 함수에 대한 수식으로 CCA와 마찬가지로 뷰를 학습한 출력물 간의 상관관계를 최대한 높이는 방향으로 학습을 한다.

$$\left(\theta_{(1)}^*, \theta_{(2)}^*\right) = \arg\max_{\theta_1,\theta_2} \operatorname{corr}\left(f_1\left(X_1;\theta_1\right), f_2\left(X_2;\theta_2\right)\right), \tag{5.2}$$

여기서 $\theta = \{\theta^{(1)}, \theta^{(2)}\} = \{W^{(1)}, W^{(2)}, b^{(1)}, b^{(2)}\}$은 DCCA 모형의 모수이다. 다른 멀티 뷰에 대한 신경망 알고리즘이 중간의 은닉층에서 뷰 간의 결합을 하는 것과는 달리 DCCA는 출력층에서 결합을 하는 것이 특징이다.

Deep GCCA (DGCCA)은 GCCA에 DCCA를 응용한 방법으로 GCCA와 같이 두 개 이상의 뷰에 대해서 학습을 시킬 수 있는 방법으로 뷰 간에 학습된 표현의 상관관계를 최대화하기 위하여 각 뷰에 대해서 DCCA와 마찬가지로 DNN 구조를 통해 비선형 매핑을 학습한다 (Benton 등, 2017).

Figure 5.2의 DGCCA의 모형을 보면 $X_j \in \mathbf{R}^{d_j \times N}$는 j번째 뷰의 데이터, N과 d_j은 j번째 뷰의 데이터의 수와 차원, r은 학습된 표현의 차원, U_j는 j번째 뷰의 선형 변환 행렬이라고 할 때, j번째 뷰의 신경망은 K_j개의 층으로 구성되어 있고 출력층은 크기 o_j이면서 c_j 갯수의 노드를 갖는다. j번째 뷰의 k번째 층은 DCCA와 같이 $h_k^j = \sigma(W_k^j h_{k-1}^j)$으로 이전 노드가 비선형 변환되어 입력된다. 최종 출력물은

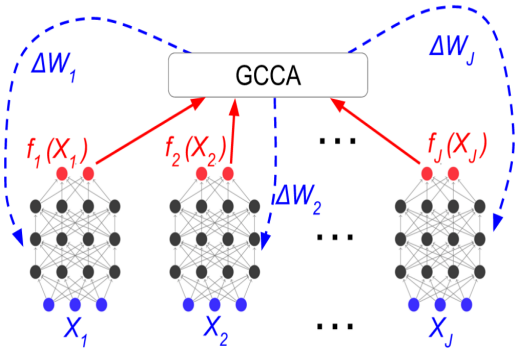

Figure 5.2. A schematic of DGCCA with deep networks for J views (Benton *et al.*, 2017).

$f_j(X_j)$로 표현된다. DGCCA는 식 (5.3)을 최소화하는 방향으로 학습한다. 여기서 $W_k^j \in \mathbf{R}^{c_k \times c_{k-1}}$는 j번째 뷰의 k번째 층의 가중치 행렬이다.

$$\min_{U_j, G} \sum_{j=1}^{J} \left|\left| G - U_j^T f(X_j) \right|\right|_F^2 \quad \text{subject to} \quad GG^T = I_r, \tag{5.3}$$

여기서 $G \in \mathbf{R}^{r \times N}$은 학습하고자 하는 공유된 표현이다.

5.2. RBM 기반 기법

멀티 모달 DBM은 RBM을 멀티 뷰 데이터 응용한 방법으로 멀티 뷰 데이터의 분포를 학습할 수 있는 생성적 모형이다. 먼저 모형의 근간이 되는 RBM은 Boltzmann Machine의 단순화된 모형으로 입력 노드끼리의 연결이 없다 (Smolensky, 1986). 다른 신경망 모형과 다른 점은 가시 변수(visible variable) $\mathbf{v} \in \{0,1\}^D$가 은닉 변수(hidden variable) $\mathbf{h} \in \{0,1\}^F$와 방향이 없이 연결되어 있고 $\mathbf{v}$, $\mathbf{h}$가 확률변수라는 점이다. 모형에 사용되는 에너지 함수 $E : \{0,1\}^D \times \{0,1\}^F$는 식 (5.4)와 같다.

$$E(\mathbf{v}, \mathbf{h}; \theta) = -\sum_{i=1}^{D}\sum_{j=1}^{F} W_{ij} v_i h_j - \sum_{i=1}^{D} b_i v_j - \sum_{j=1}^{F} a_j h_j, \tag{5.4}$$

여기서 $\theta = \{\mathbf{a}, \mathbf{b}, \mathbf{W}\}$는 RBM 모형의 모수이다. W_{ij}는 i번째 가시 변수와 j번째 은닉 변수를 연결하는 가중치이고, b_j와 a_j는 편향이다. RBM의 목적은 입력 노드와 은닉 노드의 결합 분포 혹은 조건부 분포를 학습하는 것으로 입력 노드와 은닉 노드 전체에 대한 결합 분포는 식 (5.5)와 같다.

$$P(\mathbf{v}, \mathbf{h}; \theta) = \frac{1}{Z(\theta)} \exp(-E(\mathbf{v}, \mathbf{h}; \theta)), \qquad Z(\theta) = \sum_{\mathbf{v},\mathbf{h}} \exp(-E(\mathbf{v}, \mathbf{h}; \theta)), \tag{5.5}$$

여기서 $Z(\theta)$는 정규화 상수이다. 결합 분포를 보면 가시 변수와 은닉 변수의 연결을 에너지로 취하는 항이 들어간 Boltzmann 분포임을 알 수 있다.

멀티 모달 DBM은 텍스트 데이터를 학습하는 Replicated Softmax 모형, 이미지 데이터를 학습하는 Gaussian RBM 모형, 두 모형이 학습한 표현을 결합하는 노드로 구성된다. Replicated Softmax 모형은 텍스트 데이터처럼 희박한 성질의 이산형 데이터를 학습하는데 있어서 유용한 모형이다. RBM은 가시 변수가 0이나 1을 값으로 하는 이진 항이지만 Replicated Softmax에서는 상태가 여러 개인 다항 항

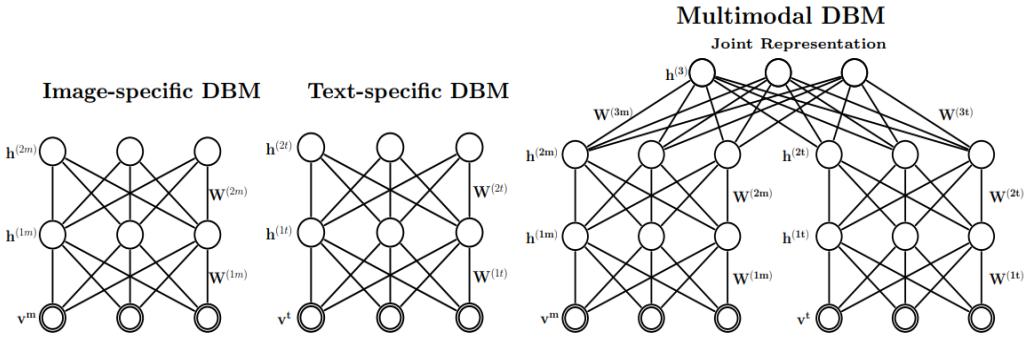

Figure 5.3. A schematic of Multimodal DBM (Srivastava and Salakhutdinov, 2012).

으로 대체된다 (Hinton과 Salakhutdinov, 2009). Replicated Softmax 모형을 이용해 텍스트 데이터의 표현을 학습하는 것의 장점은 길이가 다른 여러 개의 문서를 학습시킬 수 있고 잠재된 토픽에 대한 사후 분포를 쉽게 계산할 수 있다는 것이다. Replicated Softmax 모형의 에너지 함수는 식 (5.6)과 같다.

$$E(\mathbf{V},\mathbf{h}) = -\sum_{i=1}^{M}\sum_{j=1}^{F}\sum_{k=1}^{K} W_{ijk} v_{ik} h_j - \sum_{i=1}^{M}\sum_{k=1}^{K} b_{ik} v_{ik} - \sum_{j=1}^{F} a_j h_j, \tag{5.6}$$

여기서 K는 사전의 크기, M은 문서에 나타나는 단어의 수이고 은닉 변수 $\mathbf{h} \in \{0,1\}^F$는 RBM과 마찬가지로 이진 값을 갖는 확률변수이다. $\mathbf{V}$는 $M \times K$크기의 관측된 이진 행렬로 i번째 가시 변수가 k번째 값을 취할 때 $v_{ik} = 1$이 된다. 만약 단어의 순서를 무시하는 bags of words 가정이 적용되면 식 (5.6)은 식 (5.7)과 같이 더욱 간단해진다.

$$E(\mathbf{V},\mathbf{h}) = -\sum_{j=1}^{F}\sum_{k=1}^{K} W_{jk} \hat{v}_k h_j - \sum_{k=1}^{K} b_k \hat{v}_k - M\sum_{j=1}^{F} a_j h_j, \tag{5.7}$$

여기서 $\hat{v}_k = \sum_{i=1}^{M} v_{ik}$는 k번째 단어의 수를 뜻한다. $\hat{v}_k$가 편향 항을 스케일링 하면서 문서의 길이가 다른 경우를 고려할 수 있게 된다. Gaussian-Bernoulli RBMs는 이미지 데이터와 같은 실숫값을 갖는 데이터의 표현을 학습시킬 때 사용할 수 있는 모형이다. 가시 변수 $\mathbf{v} \in \mathbf{R}^D$는 실숫값을 갖고 은닉 변수 $\mathbf{h} \in \{0,1\}^F$는 이진값을 갖는다. Gaussian-Bernoulli RBMs의 에너지 함수는 식 (5.8)과 같이 정의될 수 있다 (Freund와 Haussler, 1992).

$$E(\mathbf{v},\mathbf{h};\theta) = \sum_{i=1}^{D}\frac{(v_i - b_i)^2}{2\sigma_i^2} - \sum_{i=1}^{D}\sum_{j=1}^{F} W_{ij} v_i h_j - \sum_{j=1}^{F} a_j h_j. \tag{5.8}$$

DBM은 RBM 모형의 은닉층의 수를 늘린 모형으로 데이터의 심층적인 분포를 학습할 수 있는 모형이다. 가시 변수 $\mathbf{v} \in \{0,1\}^F$와 은닉 변수 $\mathbf{h}^{(1)} \in \{0,1\}^{F_1}$, $\mathbf{h}^{(2)} \in \{0,1\}^{F_2}, \ldots, \mathbf{h}^{(L)} \in \{0,1\}^{F_L}$가 근접한 층에 서로 연결되는 구조를 갖는다. 은닉층이 3개인 DBM 모형의 에너지 함수는 식 (5.9)와 같다.

$$\begin{aligned} E(\mathbf{v},\mathbf{h};\theta) = &-\sum_{i=1}^{D}\sum_{j=1}^{F_1} W_{ij}^{(1)} v_i h_j^{(1)} - \sum_{j=1}^{F_1}\sum_{l=1}^{F_2} W_{jl}^{(2)} h_j^{(1)} h_l^{(2)} - \sum_{l=1}^{F_2}\sum_{p=1}^{F_3} W_{lp}^{(3)} h_l^{(2)} h_p^{(3)} \\ &- \sum_{i=1}^{D} b_j v_l - \sum_{j=1}^{F_1} b_j^{(2)} h_j^{(1)} - \sum_{l=1}^{F_2} b_l^{(2)} h_l^{(2)} - \sum_{p=1}^{F_3} b_p^{(3)} h_p^{(3)}. \end{aligned} \tag{5.9}$$

멀티 모달 DBM은 위에서 소개한 텍스트 데이터에 대해서는 Replicated Softmax 모형에 이미지 데이터에 대해서는 Gaussian-Bernoulli 모형을 이용하여 학습 하는 방법이다. Figure 5.3을 보면 각각의 모형은 DBM 구조를 취하고 마지막 은닉층에서 두 모형의 표현이 결합된다. $\mathbf{v}^m \in \mathbf{R}^D$은 실숫값을 갖는 이미지 입력 노드, $\mathbf{v}^t \in \{1, \ldots, K\}$은 M개의 단어를 포함하는 관련된 텍스트 입력 노드 그리고 v_k^t는 k번째 단어의 횟수이다. 입력 노드 전부에 대한 결합 분포는 식 (5.10)과 같다.

$$
\begin{aligned}
P(\mathbf{v}^m, \mathbf{v}; \theta) &= \sum_{\mathbf{h}^{(2m)}, \mathbf{h}^{(2t)}, \mathbf{h}^{(3)}} P\left(\mathbf{h}^{(2m)}, \mathbf{h}^{(2t)}, \mathbf{h}^{(3)}\right) \sum_{\mathbf{h}^{(1m)}} P\left(\mathbf{v}^m, \mathbf{h}^{(1m)} | \mathbf{h}^{(2m)}\right) \sum_{\mathbf{h}^{(1t)}} P\left(\mathbf{v}^t, \mathbf{h}^{(1t)} | \mathbf{h}^{(2t)}\right) \\
&= \frac{1}{Z(\theta)_M} \sum_{\mathbf{h}} \exp\left(\sum_{kj} W_{kj}^{(1t)} v_k^t h_j^{(1t)} + \sum_{jl} W_{jl}^{(2t)} h_j^{(1t)} h_j^{(2t)} + \sum_k b_k^t v_k^t + M \sum_j b_j^{(1t)} h_j^{(1t)} \right. \\
&\quad \left. + \sum_l b_l^{(2t)} h_l^{(2t)} \right) - \sum_i \frac{(v_i^m - b_i^m)^2}{2\sigma_i^2} + \sum_{ij} \frac{v_i^m}{\sigma_i} W_{ij}^{(1m)} h_j^{(1m)} + \sum_{jl} W_{jl}^{(2m)} h_j^{(1m)} h_l^{(2m)} \\
&\quad + \sum_j b_j^{(1m)} h_j^{(1m)} + \sum_l b_l^{(2m)} h_l^{(2m)} + \sum_{np} W^{(3t)} h_l^{(2t)} h_p^{(3)} + \sum_{lp} W^{(3m)} h_l^{(2m)} h_p^{(3)} \\
&\quad + \sum_p b_p^{(3)} h_p^{(3)}. \qquad (5.10)
\end{aligned}
$$

식 (5.10)을 보면 이미지 데이터를 학습할 수 있는 Gaussian-Bernoulli 모형과 텍스트 데이터를 학습할 수 있는 Replicated Softmax 모형이 공유하는 층을 연결하고 두 모형에서 추출된 특징이 더해지는 식으로 결합하는 것을 알 수 있다.

멀티 모달 DBM를 통해 학습된 표현을 이용하여 멀티 뷰 데이터의 조건부 분포를 학습하면 두 데이터의 조건부 분포를 학습할 수 있다는 장점이 있다. 예를 들어 한 뷰의 데이터가 결측 됐을 때 조건부 분포를 이용하여 다른 결측된 데이터를 샘플링하는 것이 가능하다. 또한 이미지 데이터가 주어졌을 때 조건부 분포를 이용하여 텍스트 데이터를 생성해낼 수 있다. Gibbs sampling을 이용하여 여러 뷰의 데이터의 결합 분포를 샘플링하는 것도 가능해진다. 멀티 모달 DBM을 이용하여 학습된 표현은 linear discriminant analysis (LDA), SVM 같은 지도 학습 기법을 이용하여 이미지, 텍스트의 분류 문제에 사용 가능하다. 또한 텍스트가 주어졌을 때의 이미지 검색, 이미지가 주어졌을 때 텍스트 검색과 같은 정보 검색의 문제에도 이용이 가능하다 (Hinton과 Salakhutdinov, 2009).

5.3. Autoencoder 기반 기법

Autoencoder는 비지도 기법의 신경망 모형으로 인코더에서 입력 데이터의 표현을 학습하여 디코더 부분에서 다시 입력을 복원하는 구조를 갖는다 (Bengio, 2009). Autoencoder는 데이터의 결측치 보정, 고차원 데이터의 차원 축소 등에 이용되며, PCA와는 다르게 비선형적인 표현을 학습할 수 있어 일반화된 비선형 PCA로도 볼 수 있다 (Hinton과 Salakhutdinov, 2006).

Figure 5.4의 Autoencoder 학습 구조를 보면 입력 값 $X \in \mathbf{R}^D$가 encoder에서 신경망 구조의 비선형 변환 $Z = \sigma(W_e X + b_e)$을 통해 X가 임베딩 $Z \in \mathbf{R}^K$로 매핑된다. 이후 Z에서 압축된 데이터가 비선형 변환 $\hat{X} = \sigma(W_d Z + b_d)$을 통해 Z가 $\hat{X}$로 매핑하여 X를 복원한다. Autoencoder 모형은 식 (5.11)과 같은 손실 함수를 사용하여 X와 복원된 $\hat{X}$의 복원오류(reconstruction error) 최소화하는 방향으로 학습된다 (Jaques 등, 2017).

$$
L\left(X, \hat{X}\right) = \left|\left|X - \hat{X}\right|\right|_2^2. \qquad (5.11)
$$

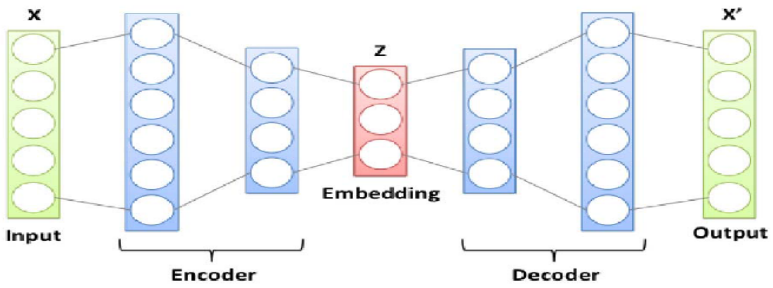

Figure 5.4. Autoencoder model (Jaques *et al.*, 2017).

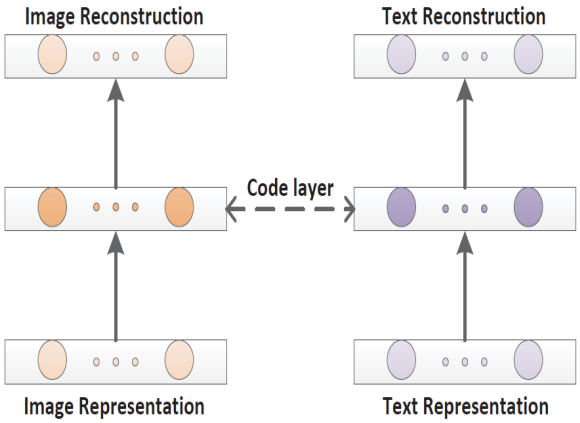

Figure 5.5. Correspondence Autoencoder (Feng *et al.*, 2014).

두 가지의 뷰가 있는 데이터에서 한 뷰의 데이터로 다른 뷰의 데이터를 검색하는 문제에 있어서 두 뷰 간의 심층적인 공유된 표현을 학습하는 것이 중요하다. 예를 들어 꽃과 하늘이 함께 나와 있는 사진이 있고 사진에 대한 태그가 "푸른", "하늘", "니콘"이라는 태그가 있다고 하자. 텍스트 뷰와 이미지 뷰 양쪽에 정보를 갖고 있는 단어는 "푸른"과 "하늘"이고 텍스트 뷰에 대해서만 정보를 갖는 단어는 "니콘", 태그에는 없지만 이미지에만 있는 정보는 "꽃"이다. 이때 한 뷰로 다른 뷰의 정보를 찾기 위해서는 양쪽 뷰에 동시에 정보를 갖는 "푸른"과 "하늘"라는 대한 표현의 상관관계를 학습시키는 것이 중요하다. 이를 위해 CCA 기법에 기반한 멀티 뷰 기법은 이 표현을 학습하는 단계와 상관관계를 학습하는 단계로 나누어서 학습하지만 correspondence autoencoder (Corr-AE)는 이 단계를 한 번으로 줄인 모형이다 (Feng 등, 2104).

Corr-AE은 Autoencoder 모형을 멀티 뷰 데이터에 적용하기 위하여 Figure 5.5와 같이 뷰마다 Autoencoder 모형을 각각 만들고 각 모형의 중간에 코드 레이어(code layer)를 만들어 서로의 상관관계를 학습시키는 모형이다. i번째 이미지 표현 $p^{(i)}$와 대응되는 텍스트 $q^{(i)}$표현 주어졌을 때 $f(p; W_f)$를 이미지 데이터를 학습하는 Autoencoder의 코드 레이어로의 매핑, $g(p; W_g)$를 텍스트 데이터를 학습하는 Autoencoder의 코드 레이어로의 매핑이라고 했을 때 둘 사이의 유사도는 식 (5.12)와 같이 측정된다.

$$C\left(p^{(i)}, q^{(i)}; W_f, W_g\right) = \left|\left|f\left(p^{(i)}; W_f\right) - g\left(q^{(i)}; W_g\right)\right|\right|_2^2 \tag{5.12}$$

여기서 f는 이미지 뷰에 대한 로지스틱 활성화 함수, g는 텍스트 뷰에 대한 로지스틱 활성화 함수이다. f와 g의 거리는 L_2노름 $||\ ||_2$으로 측정된다. Corr-AE가 이미지 표현의 입력 데이터와 텍스 표현의 입력 데이터 간의 유사한 표현을 학습시키기 위해서 식 (5.13)과 같은 손실 함수를 최소화하는 방향으로 학습된다.

$$
\begin{aligned}
L\left(p^{(i)}, q^{(i)}; \theta\right) &= (1-\alpha)\left(L_I\left(p^{(i)}; \theta\right) + L_T\left(q^{(i)}; \theta\right)\right) + \alpha L_C\left(p^{(i)}, q^{(i)}; \theta\right), \quad 0 < \alpha < 1 \\
L_I\left(p^{(i)}, q^{(i)}; \theta\right) &= \left|\left|p(i) - \hat{p}_I^{(i)}\right|\right|_2^2 \\
L_T\left(p^{(i)}, q^{(i)}; \theta\right) &= \left|\left|p(i) - \hat{p}_T^{(i)}\right|\right|_2^2 \\
L_C\left(p^{(i)}, q^{(i)}; \theta\right) &= C\left(p^{(i)}, q^{(i)}; \theta\right)
\end{aligned}
\tag{5.13}
$$

여기서 $\theta = \{W_f, W_g\}$는 각 뷰를 학습시키기 위한 가중치로, Corr-AE의 모수이다. L_I는 이미지 표현 네트워크에서의 복원오류, L_T는 텍스트 표현 네트워크에서의 복원 오류, L_C는 상관관계에 대한 손실 함수이다. 전체 손실 함수는 두 네트워크의 손실 함수 L_I와 L_T의 합과 L_C의 가중 평균으로 정해진다.

autoencoder 모형 역시 RBM과 같은 생성적 모형인 variational autoencoder (VAE)라는 모형이 있다. 이 모형은 autoencoder에 variational Bayesian 기법을 응용한 생성적인 모형이다 (Doersch, 2016). Autoencoder와 신경망 연결 구조는 비슷하지만 은닉 노드가 확률변수가 된다. VAE의 가정은 다음과 같다 (Kingma와 Welling, 2013). 사전 분포 $P(\mathbf{z})$를 따르는 확률 잠재 변수 $\mathbf{z} \sim P(\mathbf{z})$가 주어졌을 때, 관측 변수 $\mathbf{x}$는 조건부 분포 $\mathbf{x} \sim P_\theta(\mathbf{x}|\mathbf{z})$를 따른다. 여기서 θ는 분포 P의 모수이다. VAE의 학습 목표는 주변 분포(marginal distribution) $P(\mathbf{x})$를 최대화하는 것이다. 하지만 $P(\mathbf{x}) = \int P_\theta(\mathbf{x}|\mathbf{z})P(\mathbf{z})d\mathbf{z}$로 더 이상 전개하기가 어려워, 주변 분포의 하한인 evidence lower bound (ELBO) $L_{\text{VAE}}(\mathbf{x})$를 최대화하는 방법을 사용하는데 이는 식 (5.14)와 같다.

$$
\log P(\mathbf{x}) \geq L_{\text{VAE}}(\mathbf{x}), L_{\text{VAE}}(\mathbf{x}) = -D_{\text{KL}}(q_\Phi(\mathbf{z}|\mathbf{x})||P(\mathbf{z})) + E_{q_\Phi(\mathbf{x})}[\log P_\theta(\mathbf{x}|\mathbf{z})], \tag{5.14}
$$

여기서 $q_\Phi(\mathbf{z}|\mathbf{x})$는 사후분포 $P(\mathbf{z}|\mathbf{x})$의 근사분포이고, Φ는 q의 모수이다. $q_\Phi(\mathbf{z}|\mathbf{x})$는 Autoencoder의 인코더 역할을 하고 $P(\mathbf{z}|\mathbf{x})$는 디코더 역할을 한다. D_{KL}은 Kullback-Leibler divergence로 분포 $q_\Phi(\mathbf{z}|\mathbf{x})$가 $P(\mathbf{z})$로부터 발산하는 정도를 측정하는 척도로 여기서는 정규화 항의 역할을 한다 (Kullback과 Leibler, 1951). $E_{q_\Phi(\mathbf{x})}[\log P_\theta(\mathbf{x}|\mathbf{z})]$는 복원 오류이다.

Joint multimodal VAE (JMVAE)는 VAE를 멀티 뷰 학습에 응용한 방법으로 두 개의 뷰의 결합 분포 $P(\mathbf{x}, \mathbf{w})$를 모델링하는 모형이다 (Suzuki 등, 2016). JMVAE의 특징은 여러 뷰의 특징을 양 방향으로 학습할 수 있다는 것이다. 예를 들어, 이미지 뷰가 주어졌을 때 텍스트 데이터를 샘플링할 수 있고, 반대로 텍스트 뷰가 주어졌을 때 이미지 데이터를 샘플링 할 수 있다. $(\mathbf{X}, \mathbf{W}) = \{((\mathbf{x}_i, \mathbf{w}_i))\}_{i=1}^N$은 $\mathbf{x}$, $\mathbf{w}$가 각각의 뷰 1, 뷰 2인 데이터이다. JMVAE 모형은 잠재 변수 $\mathbf{z}$가 주어졌을 때 $(\mathbf{X}, \mathbf{W})$가 조건부 독립이라고 가정한다. 이때 $\mathbf{z} \sim P(\mathbf{z})$이면, 조건부 독립성에 의해 $(\mathbf{x}, \mathbf{w}) \sim P(\mathbf{x}, \mathbf{w}|\mathbf{z}) = P_{\theta_\mathbf{x}}(\mathbf{x}|\mathbf{z})P_{\theta_\mathbf{w}}(\mathbf{w}|\mathbf{z})$이 된다. 여기서 $\theta_\mathbf{x}, \theta_\mathbf{w}$은 독립적인 분포 P의 각각의 모수이다.

Figure 5.6(a)는 JMVAE의 학습 도안도로 JMVAE의 생성 과정을 표현하였다. JMVAE에서는 $P_\Phi(\mathbf{z}|\mathbf{x}, \mathbf{w})$를 인코더로 사용하고, 각 뷰가 다른 특징 표현을 갖고 있다는 점을 고려하여 각 뷰에 대하여 다른 디코더 $P_{\theta_\mathbf{x}}(\mathbf{x}|\mathbf{z})$, $P_{\theta_\mathbf{w}}(\mathbf{w}|\mathbf{z})$를 사용한다. 뷰 2가 소실 된 상황에 가정하면, 학습된 JMVAE는 뷰 1의 데이터 $\mathbf{x}$가 입력되고(회색 원), $\mathbf{w}$는 0으로 고정된다(흰색 원). 이후 decoder $P_{\theta_\mathbf{w}}(\mathbf{w}|\mathbf{z})$는 뷰 2를 샘플링한다. 하지만 소실된 뷰의 데이터가 고차원의 복잡한 데이터이면 잠재 변수 $\mathbf{z}$가 불완전해져 샘플링이

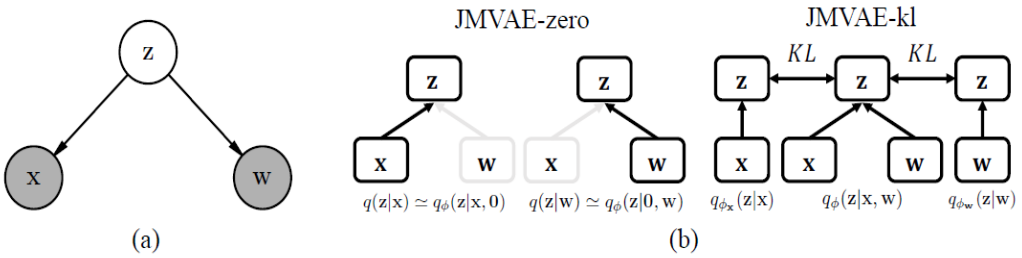

Figure 5.6. (a) Graphical model of JMVAE (b) Two approaches to estimate encoders with a single input, $q_{\Phi_\mathbf{x}}(\mathbf{z}|\mathbf{x})$, $q_{\Phi_\mathbf{w}}(\mathbf{z}|\mathbf{x})$ on the JMVAE (Suzuki *et al.*, 2016).

제대로 안 될 수 있다. JMVAE의 $\log P(\mathbf{x}, \mathbf{w})$에 대한 ELBO $L_{\mathrm{JM}}(\mathbf{x}, \mathbf{w})$은 식 (5.15)와 같다.

$$\begin{aligned} L_{\mathrm{JM}}(\mathbf{x}, \mathbf{w}) &= E_{q_\Phi(\mathbf{z}|\mathbf{x},\mathbf{w})}\left(\log \frac{P_\theta(\mathbf{x}, \mathbf{w}, \mathbf{z})}{q_\Phi(\mathbf{z}|\mathbf{x}, \mathbf{w})}\right) \\ &= -D_{\mathrm{KL}}\left(q_\Phi(\mathbf{z}|\mathbf{x}, \mathbf{w})||P(\mathbf{z})\right) + E_{(q_\Phi(\mathbf{z}|\mathbf{x},\mathbf{w}))}[\log P_{\theta_\mathbf{x}}(\log P_{\theta_\mathbf{w}}(\mathbf{x}|\mathbf{z}))] \\ &\quad + E_{(q_\Phi(\mathbf{z}|\mathbf{x},\mathbf{w}))}[\log P_{\theta_\mathbf{w}}(\log P_{\theta_\mathbf{w}}(\mathbf{w}|\mathbf{z}))] \end{aligned} \tag{5.15}$$

Suzuki 등 (2018)은 JMVAE를 JMVAE-kl로 개선하여 샘플링이 제대로 안될 수 있는 문제를 고려하였다. Figure 5.6(b)은 JMVAE-kl의 학습 과정이다. $q_{\Phi_\mathbf{x}}(\mathbf{z}|\mathbf{x})$, $q_{\Phi_\mathbf{w}}(\mathbf{z}|\mathbf{x})$은 각각의 뷰에 대한 인코더로 Kullback-Leibler divergence를 통해 인코더 $q_\Phi(\mathbf{z}|\mathbf{x}, \mathbf{w})$에 가까워지는 방향으로 학습된다. 여기서 $\Phi_\mathbf{x}$, $\Phi_\mathbf{w}$은 모형의 모수이다. JMVAE-kl의 목적 함수 $L_{\mathrm{JM_{kl}}(\alpha)}(\mathbf{x}, \mathbf{w})$는 식 (5.16)과 같다.

$$L_{\mathrm{JM_{kl}}(\alpha)}(\mathbf{x}, \mathbf{w}) = L_{\mathrm{JM}}(\mathbf{x}, \mathbf{w}) - \alpha(D_{\mathrm{KL}}(q_\Phi(\mathbf{z}|\mathbf{x}, \mathbf{w})||q_{\Phi_\mathbf{x}}(\mathbf{z}|\mathbf{x})) + q_\Phi(\mathbf{z}|\mathbf{x}, \mathbf{w})||q_{\Phi_\mathbf{w}}(\mathbf{z}|\mathbf{w})), \tag{5.16}$$

여기서 α는 Kullback-Leibler divergence를 조절하는 항이다.

5.4. GAN 기반 기법

Generative adverserial network (GAN)은 RBM, VAE와 같은 생성적 모형으로 데이터를 생성할 수 있는 모형이다. GAN의 학습 구조를 보면 두 개의 경기 참여자 생성자 G와 식별자 D가 미니맥스(minimax) 게임을 하면서 서로 경쟁하는 구조이다. 식별자는 합성 이미지(synthetic image)로부터 실제 훈련 데이터를 구별하려 하고 생성자는 식별자를 속이려 한다. GAN을 비유적으로 설명하면 생성자는 위조지폐범으로 가짜 화폐를 계속 만들고 식별자는 위조지폐를 식별해내는 경찰이다. 위조지폐범이 만들어내는 가짜 화폐는 처음에는 형편없지만 경찰과의 적대적 경쟁 속에서 점점 진짜 화폐와 비슷하게 된다 (Goodfellow 등, 2014). GAN 모형의 생성자와 식별자의 경쟁을 수식적으로 표현하면 식 (5.17)과 같다.

$$\min_G \max_D V(D, G) = E_{x\sim P_{\mathrm{data}}(x)}[\log D(x)] + E_{z\sim P_z(z)}[\log(1 - D(G(z)))], \tag{5.17}$$

여기서 $p_z(z)$는 입력 노이즈 변수에 대한 사전 분포이고, $G(z)$는 MLP에 대한 함수이고, $D(x)$는 스칼라 값을 출력하는 MLP에 대한 함수로 x가 P_g가 아닌 데이터로부터 왔을 확률이다. D는 훈련 데이터와 G로부터 나온 샘플을 정확하게 분류할 확률이 최대가 되도록 학습되고 G는 반대로 $\log(1 - D(G(z)))$가 최소화 되도록 학습된다.

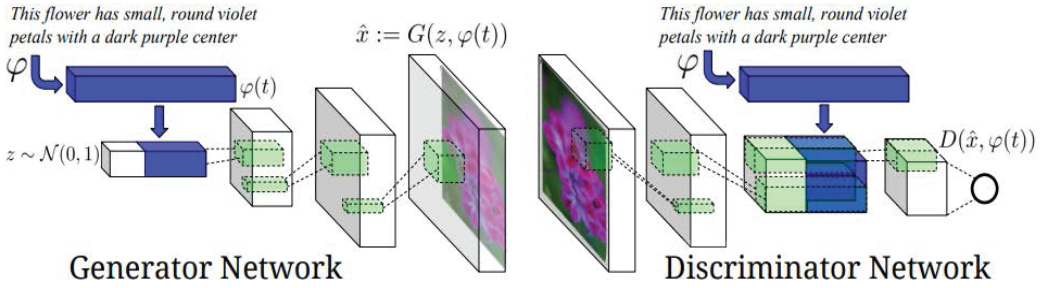

Figure 5.7. Architecture of text-conditional convolutional GAN (Reed *et al.*, 2016).

Reed 등 (2016)은 DC-GAN 모형 (Radford 등, 2015)을 멀티 뷰 학습에 응용하여 텍스트가 주어졌을 때 연관된 이미지를 생성하는 방법을 연구하였다. 먼저 텍스트의 특징을 인코더 $\varphi(t)$로 임베딩 한다. 이때 $\varphi(t)$는 CNN이나 RNN을 사용하고 생성자와 식별자 모두에게 사용된다. 생성자의 네트워크는 $G : \mathbf{R}^Z \times \mathbf{R}^T \to \mathbf{R}^D$이고, 식별자의 네트워크는 $D : \mathbf{R}^D \times \mathbf{R}^T \to \{0, 1\}$이다. 여기서 T는 텍스트 임베딩의 차원, D는 이미지의 차원이며 Z는 G로 입력되는 노이즈 $z \in \mathbf{R}^Z \sim N(0, 1)$의 차원이다.

Figure 5.7은 GAN의 학습 모형이다. 먼저 가짜 이미지 $\hat{x}$를 생성하는 과정 $G(z, \varphi(t)) \to \hat{x}$을 보면, 생성자 네트워크에 임베딩된 텍스트와 사전 분포인 정규분포에서 샘플링된 노이즈 입력값 $z \in \mathbf{R}^Z \sim N(0, 1)$가 입력층으로 들어간다. 텍스트 인코딩 $\varphi(t)$은 fully-connected layer를 통해 압축되고 디컨볼루션(deconvolution) 네트워크를 통해 이미지 $\hat{x}$를 생성한다. 이후 식별자는 생성된 이미지 $\hat{x}$를 입력하여 고해상도의 이미지 특징을 학습하기 위해 모든 컨볼루션 레이어마다 배치 정규화(batch normalization)를 한다 (Ioffe와 Szegedy, 2015). 컨볼루션 단계의 중간에 임베딩된 텍스트 $\varphi(t)$가 입력되고, 다시 fully-connected layer를 통해 압축되는 과정을 거친다. 최종적으로 $D(\hat{x}, \varphi)$에서 최종 연관 점수를 출력한다.

위 방법의 문제점은 더 높은 해상도의 이미지를 생성하기 어렵다는 것으로 Reed 등 (2016)은 64×64크기의 픽셀 이미지에 대하여 실험을 성공하였다. 이에 Zhang 등 (2017)은 GAN의 생성 단계를 단계별로 쌓아가는 식으로 모형을 개선한 StackGAN 방법을 제시하여 256×256 픽셀 이미지를 생성하는데 성공하였다.

6. 결론

최근 중요하게 연구 되고 있는 멀티 뷰 기법은 인간 행동 인식 분야, 의학 분야, 정보 검색 분야, 표정 인식 분야 등 다양한 분야에서 여러 가지 문제들을 해결하고 있다. 멀티 뷰 기법의 학습 방식은 기존의 단일 뷰의 데이터를 학습하는 방식과는 달리 여러 가지 뷰의 정보를 보완적으로 이용하여 단일 뷰의 데이터를 학습시킬 때보다 더 좋은 결과를 내는 것을 목표로 한다. 본 연구에서는 이러한 멀티 뷰 기법의 목표를 달성하기 위해 멀티 뷰 기법이 어떠한 원리로 데이터를 학습하고 데이터를 통합하는지에 대하여 기술하였다. 또한 멀티 뷰 기법의 데이터 통합 방식을 쉽게 이해시키기 위하여 통합 방식을 데이터 차원, 분류기 차원, 학습된 표현의 차원으로 나누어 멀티 뷰 기법의 학습 원리를 이해하기 쉽게 소개하고자 하였다.

딥 러닝 기법은 이미지, 영상, 음성 등 다양한 데이터의 학습에서 좋은 성능을 보이고 있어 최근 기계학습 분야에서 광범위하게 사용되고 있다. 딥 러닝 기법 역시 멀티 뷰 학습에 응용되어 다양한 방식으

Table 6.1. R & Python Implementations of Multi-view Learning Models

멀티 뷰 기법	모형	언어	패키지	비고
데이터 차원의 통합	CCA	R	CCA	
		R	stats	
		Python	sklearn	
	KCCA	R	kernlab	
		Python	pyrcca	https://github.com/gallantlab/pyrcca
	GCCA	R	drCCA	
		R	RGCCA	
		Python	numpy 기반	https://github.com/rupy/GCCA
분류기 차원의 통합	Co-training	R	SSL	
		Python	sklearn 기반	https://github.com/jjrob13/sklearn cotraining/blob/master/sklearn_cotraining/classifiers.py
학습된 표현 간의 통합	Seq2seq	Python	Tensorflow 기반	
	m-RNN	Python	Tensorflow 기반	http://www.stat.ucla.edu/~junhua.mao/m-RNN.html
	Deep CCA	Python	Tensorflow 기반	https://github.com/VahidooX/DeepCCA
		Python	Theano 기반	https://github.com/msamribeiro/deep-ccaA
	multimodal DBM	Python	Tensorflow 기반	https://github.com/abyoussef/DRBM_Project
	Corr-AE	Python	numpy 기반	https://github.com/huyt16/Twitter100k
	JMVAE	Python	Theano 기반	https://github.com/masa-su/jmvae
	Text-Image GAN	Python	Tensorflow 기반	https://github.com/paarthneekhara/text-to-x image

로 멀티 뷰 데이터를 통합하는데 성공하였다. 본 연구에서는 딥 러닝 기법에서 가장 많이 쓰이는 기법인 CNN, RNN, RBM, Autoencoder, GAN에 기반한 기법들이 멀티 뷰 기법에 어떻게 응용되고 있는지 살펴보았다. 또한 이 기법들 중 CNN, RNN 기반 멀티 뷰 기법들은 기계학습 방법론 중 지도학습으로 분류하고 RBM, Autoencoder, GAN에 기반한 멀티 뷰 기법들은 비지도 학습으로 분류하여 이 기법들이 멀티 뷰 기법에 어떠한 방식으로 사용되는지 이해를 돕고자 하였다.

Table 6.1은 본 연구에서 소개했던 멀티 뷰 기법들이 R 혹은 Python으로 구현되어 있는 것을 정리한 결과로 실제 분석을 하는데 있어 도움이 되고자 하였다. 대부분의 Python으로 구현된 기법들은 패키지 안에 구현되어 있기 보다는 대표적인 패키지를 기반으로 구현되어 있는 것이 많아 이러한 경우 무슨 기반인지 명시하였다.

딥 러닝 기반 모형은 역시 다른 기계학습 모형들과 마찬가지로 완벽하지 않고 여러 가지 문제점들을 안고 있다. 멀티 뷰 기법에 응용된 딥 러닝 기법들 역시 이와 비슷한 문제들을 안고 있다. 추후 연구에서는 멀티 뷰 기법의 응용된 딥 러닝 기법들이 문제점들을 어떤 방식으로 해결할 수 있을지 연구하고자 한다. 먼저 딥 러닝 모형은 수많은 하이퍼 파라미터가 존재하게 되는데 멀티 뷰 기법에서는 일반적인 딥 러닝 기법보다 훨씬 많은 하이퍼 파라미터가 존재 하게 된다. 이 수많은 하이퍼 파라미터를 효과적으로 설정할 수 있는 방안을 연구하고자 한다. 또한 딥 러닝 알고리즘은 수많은 가중치 파라미터를 갖고 있어 훈련 데이터에 과적합할 위험이 있다. 과적합을 방지하기 위해 목적 함수에 모형 별로 정규화 항을 넣는 방법도 있고 연결 노드를 랜덤하게 드롭아웃 방법 등 여러 가지 방법이 있다. 추후 연구에서는 멀티 뷰 모형에서는 어떠한 방식의 정규화항이 적절한지 연구하고자 한다.

딥 러닝 분야는 최근 빠르게 변하고 있고 딥 러닝에서 최근 중요하게 떠오르고 있는 전이학습, 강화학습

등의 분야 역시 간과할 수 없는 중요한 분야로 위에서 소개한 기법들과 접목되고 있다. 또한 딥 러닝 기법에도 베이지안 딥 러닝, 앙상블 딥 러닝 등 새로운 방식의 기법들이 출현하고 있다. 추후 연구에서는 이러한 다양한 분야의 최신의 기법들을 탐색해 보고 멀티 뷰 데이터에 대한 응용 가능성에 대하여 탐색하고자 한다.

본 연구에서는 전통적인 멀티 뷰 기법부터 최근 많이 연구되고 있는 딥 러닝 기반 멀티 뷰 기법까지 다양한 멀티 뷰 기법에 대하여 소개하였다. 이전의 연구에서는 전통적인 멀티 뷰 기법을 다룬 연구는 딥 러닝 기법에 대한 소개가 부족하고 최근의 딥 러닝 기법을 다룬 연구에서는 전통적인 멀티 뷰 기법과의 연결성이 부족하였다. 본 연구에서는 전통적인 멀티 뷰 기법에서 최근의 멀티 뷰 기법까지 다양한 기법이 어떠한 분야에서 사용되고 있는지를 리뷰 하여 기존 연구와의 차별성을 두었다. 본 연구가 멀티 뷰 기법을 필요로 하는 인간 행동 인식 분야, 정보 검색 분야, 의학 분야 등 다양한 분야에 도움이 되기를 바란다.

References

Adetiba, E. and Olugbara, O. O. (2015). Lung cancer prediction using neural network ensemble with histogram of oriented gradient genomic features, *The Scientific World Journal*.

Andrew, G., Arora, R., Bilmes, J., and Livescu, K. (2013). Deep canonical correlation analysis, *In International Conference on Machine Learning*, 1247–1255.

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate, arXiv preprint arXiv:1409.0473.

Bai, B., Weston, J., Grangier, D., Collobert, R., Sadamasa, K., Qi, Y., and Weinberger, K. (2010). Learning to rank with (a lot of) word features, *Information retrieval*, **13**, 291–314.

Bengio, Y. (2009). Learning deep architectures for AI, *Foundations and trends㎛ℓkPa in Machine Learning*, **2**, 1–127.

Benton, A., Khayrallah, H., Gujral, B., Reisinger, D., Zhang, S., and Arora, R. (2017). Deep generalized canonical correlation analysis, arXiv preprint arXiv:1702.02519.

Blum, A. and Mitchell, T. (1998). Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory*, **11**, 92–100.

Bokhari, M. U. and Hasan, F. (2013). Multimodal information retrieval: Challenges and future trends, *International Journal of Computer Applications*, **74(14)**.

Cai, J., Tang, Y., and Wang, J. (2016). Kernel canonical correlation analysis via gradient descent, *Neurocomputing*, **182**, 322–331.

Cho, K., Van Merrienboer, B., Bahdanau, D., and Bengio, Y. (2014a). On the properties of neural machine translation: Encoder-decoder approaches, arXiv preprint arXiv:1409.1259.

Cho, K., Van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014b). Learning phrase representations using RNN encoder-decoder for statistical machine translation, arXiv preprint arXiv:1406.1078.

Cristianini, N., Shawe-Taylor, J., and Lodhi, H. (2002). Latent semantic kernels, *Journal of Intelligent Information Systems*, **18**, 127–152.

Dasgupta, S., Littman, M. L., and McAllester, D. A. (2002). PAC generalization bounds for co-training. In *Advances in Neural Information Processing Systems*, 375–382.

Dey, A. (2016). Machine learning algorithms: a review, *(IJCSIT) International Journal of Computer Science and Information Technologies*, **7**, 1174–1179.

Ding, C. and Tao, D. (2015). Robust face recognition via multimodal deep face representation, *IEEE Transactions on Multimedia*, **17**, 2049–2058.

Doersch, C. (2016). Tutorial on variational autoencoders, arXiv preprint arXiv:1606.05908.

Dou, Q., Chen, H., Yu, L., Qin, J., and Heng, P. A. (2017). Multilevel contextual 3-D CNNs for false positive reduction in pulmonary nodule detection, *IEEE Transactions on Biomedical Engineering*, **64**, 1558–1567.

Du, J., Ling, C. X., and Zhou, Z. H. (2011). When does cotraining work in real data?, *IEEE Transactions on Knowledge and Data Engineering*, **23**, 788–799.

Elman, J. L. (1990). Finding structure in time, *Cognitive science*, **14**, 179–211.

Farquhar, J., Hardoon, D., Meng, H., Shawe-Taylor, J. S., and Szedmak, S. (2006). Two view learning: SVM-2K, theory and practice. In *Advances in Neural Information Processing Systems*, 355–362.

Feng, F., Wang, X., and Li, R. (2014). Cross-modal retrieval with correspondence autoencoder. In *Proceedings of the 22nd ACM international conference on Multimedia*, (pp. 7–16), ACM

Freund, Y. and Haussler, D. (1992). Unsupervised learning of distributions on binary vectors using two layer networks. In *Advances in neural information processing systems*, 912–919.

Frome, A., Corrado, G. S., Shlens, J., Bengio, S., Dean, J., and Mikolov, T. (2013). Devise: a deep visual-semantic embedding model. In *Advances in Neural Information Processing Systems*, 2121–2129.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., and Bengio, Y. (2014). Generative adversarial nets. In *Advances in Neural Information Processing Systems*, **11**, 2672–2680.

Hardoon, D. R., Szedmak, S., and Shawe-Taylor, J. (2004). Canonical correlation analysis: an overview with application to learning methods, *Neural Computation*, **16**, 2639–2664.

Hinton, G. E. and Salakhutdinov, R. R. (2006).Reducing the dimensionality of data with neural networks, *Science*, **313(5786)**, 504–507.

Hinton, G. E. and Salakhutdinov, R. R. (2009). Replicated softmax: an undirected topic model. In *Advances in Neural Information Processing Systems*, 1607–1614.

Horst, P. (1961). Generalized canonical correlations and their applications to experimental data, *Journal of Clinical Psychology*, **17**, 331–347.

Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift., arXiv preprint arXiv:1502.03167.

Jain, A., Zamir, A. R., Savarese, S., and Saxena, A. (2016). Structural-RNN: Deep learning on spatio-temporal graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5308–5317.

Jaques, N., Taylor, S., Sano, A., and Picard, R. (2017). Multimodal Autoencoder: A Deep Learning Approach to Filling in Missing Sensor Data and Enabling Better Mood Prediction. In *Proceedings of International Conference on Affective Computing and Intelligent Interaction (ACII)*, San Antonio, Texas.

Jeni, L. A., Girard, J. M., Cohn, J. F., and De La Torre, F. (2013). Continuous au intensity estimation using localized, sparse facial feature space. In *2013 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, 1–7.

Kang, G., Liu, K., Hou, B., and Zhang, N. (2017). 3D multi-view convolutional neural networks for lung nodule classification, *PLoS One*, **12(11)**.

Kingma, D. P. and Welling, M. (2013). Auto-encoding variational Bayes, arXiv preprint arXiv:1312.6114.

Kiros, R., Popuri, K., Cobzas, D., and Jagersand, M. (2014). Stacked multiscale feature learning for domain independent medical image segmentation. In *International Workshop on Machine Learning in Medical Imaging*, 25–32. Springer, Cham.

Kullback, S. and Leibler, R. A. (1951). On information and sufficiency, *The Annals of Mathematical Statistics*, **22**, 79–86.

Kumari, J., Rajesh, R., and Pooja, K. M. (2015). Facial expression recognition: a survey, *Procedia Computer Science*, **58**, 486–491.

Lahat, D., Adali, T., and Jutten, C. (2015). Multimodal data fusion: an overview of methods, challenges, and prospects, *Proceedings of the IEEE*, **103**, 1449–1477.

Li, Y., Yang, M., and Zhang, Z. (2016). Multi-view representation learning: a survey from shallow methods to deep methods, arXiv preprint arXiv:1610.01206.

Liu, J., Jiang, Y., Li, Z., Zhou, Z. H., and Lu, H. (2015a). Partially shared latent factor learning with multiview data, *IEEE Transactions on Neural Networks and Learning Systems*, **26**, 1233–1246.

Liu, S., Liu, S., Cai, W., Che, H., Pujol, S., Kikinis, R., and Fulham, M. J. (2015b). Multimodal neuroimaging feature learning for multiclass diagnosis of Alzheimer's disease, *IEEE Transactions on Biomedical Engineering*, **62**, 1132–1140.

Lorincz, A., Jeni, L., Szabo, Z., Cohn, J., and Kanade, T. (2013). Emotional expression classification

using time-series kernels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 889–895

Ma, L., Lu, Z., Shang, L., and Li, H. (2015). Multimodal convolutional neural networks for matching image and sentence. In *Proceedings of the IEEE International Conference on Computer Vision*, 2623–2631.

Mao, J., Xu, W., Yang, Y., Wang, J., Huang, Z., and Yuille, A. (2014). Deep captioning with multimodal recurrent neural networks (m-rnn), arXiv preprint arXiv:1412.6632.

Neverova, N., Wolf, C., Taylor, G., and Nebout, F. (2016). Moddrop: adaptive multi-modal gesture recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **38**, 1692–1706.

Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., and Ng, A. Y. (2011). Multimodal deep learning. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, 689–696.

Nigam, K. and Ghani, R. (2000). Analyzing the effectiveness and applicability of co-training. In *Proceedings of the Ninth International Conference on Information and Knowledge Management*, 86–93.

Pohl, C., Ali, R. M., Chand, S. J. H., Tamin, S. S., Nazirun, N. N. N., and Supriyanto, E. (2014). Interdisciplinary approach to multimodal image fusion for vulnerable plaque detection, *In Biomedical Engineering and Sciences (IECBES), 2014 IEEE Conference on*, 11–16.

Qi, C. R., Su, H., Nießner, M., Dai, A., Yan, M., and Guibas, L. J. (2016). Volumetric and multi-view cnns for object classification on 3d data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5648–5656.

Radford, A., Metz, L., and Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks, arXiv preprint arXiv:1511.06434.

Ramachandram, D. and Taylor, G. W. (2017). Deep multimodal learning: a survey on recent advances and trends, *IEEE Signal Processing Magazine*, **34**, 96–108.

Reed, S., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., and Lee, H. (2016). Generative adversarial text to image synthesis, arXiv preprint arXiv:1605.05396.

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. Neural networks, *Social Service Review*, **61**, 85–117.

Shen, D., Wu, G., and Suk, H. I. (2017). Deep learning in medical image analysis, *Annual Review of Biomedical Engineering*, **19**, 221–248.

Sindhwani, V., Niyogi, P., and Belkin, M. (2005). A co-regularization approach to semi-supervised learning with multiple views. In *Proceedings of ICML Workshop on Learning with Multiple Views*, 74–79, Citeseer.

Sitová, Z., Šeděnka, J., Yang, Q., Peng, G., Zhou, G., Gasti, P., and Balagani, K. S. (2016). HMOG: New behavioral biometric features for continuous authentication of smartphone users *IEEE Transactions on Information Forensics and Security*, **11(5)**, 877–892.

Smolensky, P. (1986). Information processing in dynamical systems: Foundations of harmony theory (No. CU-CS-321-86), *COLORADO UNIV AT BOULDER DEPT OF COMPUTER SCIENCE*.

Sousa, R. T. and Gama, J. (2017). Comparison between Co-training and Self-training for single-target regression in data streams using AMRules.

Srivastava, N. and Salakhutdinov, R. R. (2012). Multimodal learning with deep Boltzmann machines, *In Advances in neural information processing systems*, 2222–2230.

Su, H., Maji, S., Kalogerakis, E., and Learned-Miller, E. (2015). Multi-view convolutional neural networks for 3d shape recognition. In *In Proceedings of the IEEE International Conference on Computer Vision*, 945–953.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*, 3104–3112.

Suzuki, M., Nakayama, K., and Matsuo, Y. (2016). Joint multimodal learning with deep generative models, arXiv preprint arXiv:1611.01891.

Suzuki, M., Nakayama, K., and Matsuo, Y. (2018). Improving Bi-directional Generation between Different Modalities with Variational Autoencoders, arXiv preprint arXiv:1801.08702.

Tatulli, E. and Hueber, T. (2017). Feature extraction using multimodal convolutional neural networks for visual speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, 2971–2975.

Taylor, G. W., Fergus, R., LeCun, Y., and Bregler, C. (2010). Convolutional learning of spatio-temporal

features. In *European conference on computer vision*, (pp. 140–153), Springer, Berlin, Heidelberg.

Uludağ, K. and Roebroeck, A. (2014). General overview on the merits of multimodal neuroimaging data fusion, *Neuroimage*, **102**, 3–10.

Vargas, R., Mosavi, A., and Ruiz, L. (2017). Deep Learning: A Review,*Advances in Intelligent Systems and Computing*.

Vrigkas, M., Nikou, C., and Kakadiaris, I. A. (2015). A review of human activity recognition methods, *Frontiers in Robotics and AI*, **2**, 28.

Wang, W., Ooi, B. C., Yang, X., Zhang, D., and Zhuang, Y. (2014). Effective multi-modal retrieval based on stacked auto-encoders. In *Proceedings of the VLDB Endowment*, **7**, 649–660.

Xu, C., Tao, D., and Xu, C. (2013). A survey on multi-view learning, arXiv preprint arXiv:1304.5634.

Yu, Y., Lin, H., Meng, J., Wei, X., Guo, H., and Zhao, Z. (2017). Deep transfer learning for modality classification of medical images, *Information*, **8**, 91.

Zhang, H., Xu, T., Li, H., Zhang, S., Huang, X., Wang, X., and Metaxas, D. (2017). StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks, arXiv preprint.

Zhang, W., Zhang, Y., Ma, L., Guan, J., and Gong, S. (2015). Multimodal learning for facial expression recognition, *Pattern Recognition*, **48**, 3191–3202.

Zhao, J., Xie, X., Xu, X., and Sun, S. (2017). Multi-view learning overview: recent progress and new challenges, *Information Fusion*, **38**, 43–54.

Zhou, Z. H. and Li, M. (2005). Tri-training: exploiting unlabeled data using three classifiers, *IEEE Transactions on knowledge and Data Engineering*, **86**, 660–689.

Zhu, X. (2006). Semi-supervised learning literature survey, *Computer Science*, University of Wisconsin-Madison, **2**, 4.

Zhu, X. and Goldberg, A. B. (2007). Introduction to semi-supervised learning, *Synthesis Lectures on Artificial Intelligence and Machine Learning*, **3**, 1–130.

멀티 뷰 기법 리뷰: 이해와 응용

배강일[a] · 이영섭[b] · 임창원[a,1]

[a]중앙대학교 응용통계학과, [b]동국대학교 통계학과

(2018년 11월 6일 접수, 2018년 12월 13일 수정, 2019년 1월 3일 채택)

요 약

멀티 뷰 기법은 데이터를 다양한 관점에서 보려는 접근 방법이며 데이터의 다양한 정보를 통합하여 사용하려는 시도이다. 최근 많은 연구가 진행되고 있는 멀티 뷰 기법에서는 단일 뷰 만을 이용하여 모형을 학습시켰을 때 보다 좋은 성과를 보인 경우가 많았다. 멀티 뷰 기법에서 딥 러닝 기법의 도입으로 이미지, 텍스트, 음성, 영상 등 다양한 분야에서 좋은 성과를 보였다. 본 연구에서는 멀티 뷰 기법이 인간 행동 인식, 의학, 정보 검색, 표정 인식 분야에서 직면한 여러 가지 문제들을 어떻게 해결하고 있는지 소개하였다. 또한 전통적인 멀티 뷰 기법들을 데이터 차원, 분류기 차원, 표현 간의 통합으로 분류하여 멀티 뷰 기법의 데이터 통합 원리를 리뷰 하였다. 마지막으로 딥 러닝 기법 중 가장 범용적으로 사용되고 있는 CNN, RNN, RBM, Autoencoder, GAN 등이 멀티 뷰 기법에 어떻게 응용되고 있는지를 살펴보았다. 이때 CNN, RNN 기반 학습 모형을 지도학습 기법으로, RBM, Autoencoder, GAN 기반 학습 모형을 비지도 학습 기법으로 분류하여 이 방법들이 대한 이해를 돕고자 하였다.

주요용어: 멀티 뷰 학습, 딥 러닝, 기계학습, 데이터 통합

이 논문은 2017년도 정부(과학기술정보통신부)의 재원으로 한국연구재단-차세대정보 컴퓨팅기술개발사업의 지원을 받아 수행된 연구임 (NRF-2017M3C4A7083281).

[1]교신저자: (06974) 서울특별시 동작구 흑석로 84, 중앙대학교 응용통계학과. E-mail: clim@cau.ac.kr