심층신경망 언어이해에서의 벡터-그래프 변환 방법을 통한 설명가능성 확보에 대한 연구

허세훈○, 정상근*
충남대학교 컴퓨터공학과
sehunhu5247@gmail.com, hugman@cnu.ac.kr

Vector2graph : A Vector-to-Graph Conversion Framework for Explainable Deep Natural Language Understanding

Se-Hun Hu○, Sangkeun Jung*
Department of Computer Science & Engineering, Chungnam National University

요 약

딥러닝(Deep-learning) 기반의 자연어 이해(Natural Language Understanding) 기술들은 최근에 상당한 성과를 성취했다. 하지만 딥러닝 기반의 자연어 이해 기술들은 내적인 동작들과 결정에 대한 근거를 설명하기 어렵다. 본 논문에서는 벡터를 그래프로 변환함으로써 신경망의 내적인 의미 표현들을 설명할 수 있도록 한다. 먼저 인간과 기계 모두가 이해 가능한 표현방법의 하나로 그래프를 주요 표현방법으로 선택하였다. 또한 그래프의 구성요소인 노드(Node) 및 엣지(Edge)의 결정을 위한 Element-Importance Inverse-Semantic-Importance(EI-ISI) 점수와 Element-Element-Correlation(EEC) 점수를 심층신경망의 훈련방법 중 하나인 드랍아웃(Dropout)을 통해 계산하는 방법을 제안한다. 다양한 실험들을 통해, 본 연구에서 제안한 벡터-그래프(Vector2graph) 변환 프레임워크가 성공적으로 벡터의 의미정보를 유지하면서도, 설명가능한 그래프를 생성함을 보인다. 더불어, 그래프 기반의 새로운 시각화 방법을 소개한다.

주제어: 그래프, 벡터, 드랍아웃, 설명 가능성

1. 서론

딥러닝(Deep-learning) 기반의 자연어 이해 분야는 최근 혁혁한 성과들을 보여주었다[1, 2, 3]. 그러나, 심층신경망(Deep Neural Network) 접근법들이 우수한 성능을 보이고는 있으나, 신경망 내부가 고차원 벡터들의 복잡한 연결로 구성되기 때문에, 모델 내부 처리 과정과 예측 결과들에 대해 설명하는 것은 상대적으로 매우 어렵다. 이러한 설명가능성 부재는 자연어 이해를 포함한 딥러닝 기반 자연어 처리 기술의 오류분석 및 유지관리의 비효율성을 증가시키며, 결과적으로 빠른 오류수정이 필요한 상업제품들에의 적극적인 적용을 어렵게 한다.

기존의 연구들은 모델 내부의 상태를 사람이 이해할 수 있는 형태의 데이터 모달리티(Modality)로 변환하는 방식을 통해, 설명력을 확보하고자 하였다. 몇몇 연구들[4, 5]은 모달리티로써 설명력 있는 텍스트들을 생성한다. 반면 다른 연구들[6, 7, 8]은 히트맵(Heatmap), 색상(Color), 이미지 패치(Image Patch)와 같은 시각적 마커(Visual Marker)를 제공한다.

본 논문은 설명 모달리티로써 심층신경망 내부의 벡터를 통해 변환되는 **그래프(Graph)**의 사용을 제안한다.

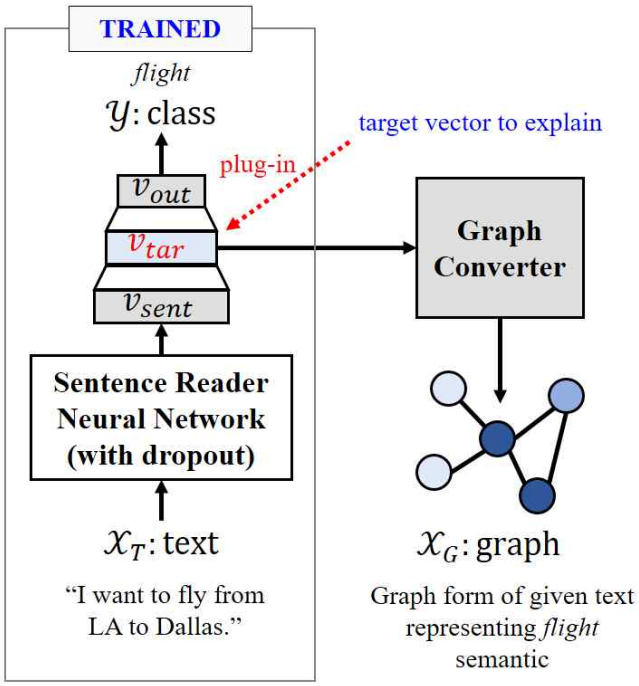

그림 1 벡터-그래프 변환 프레임워크의 구조

* 교신 저자(Corresponding Author)

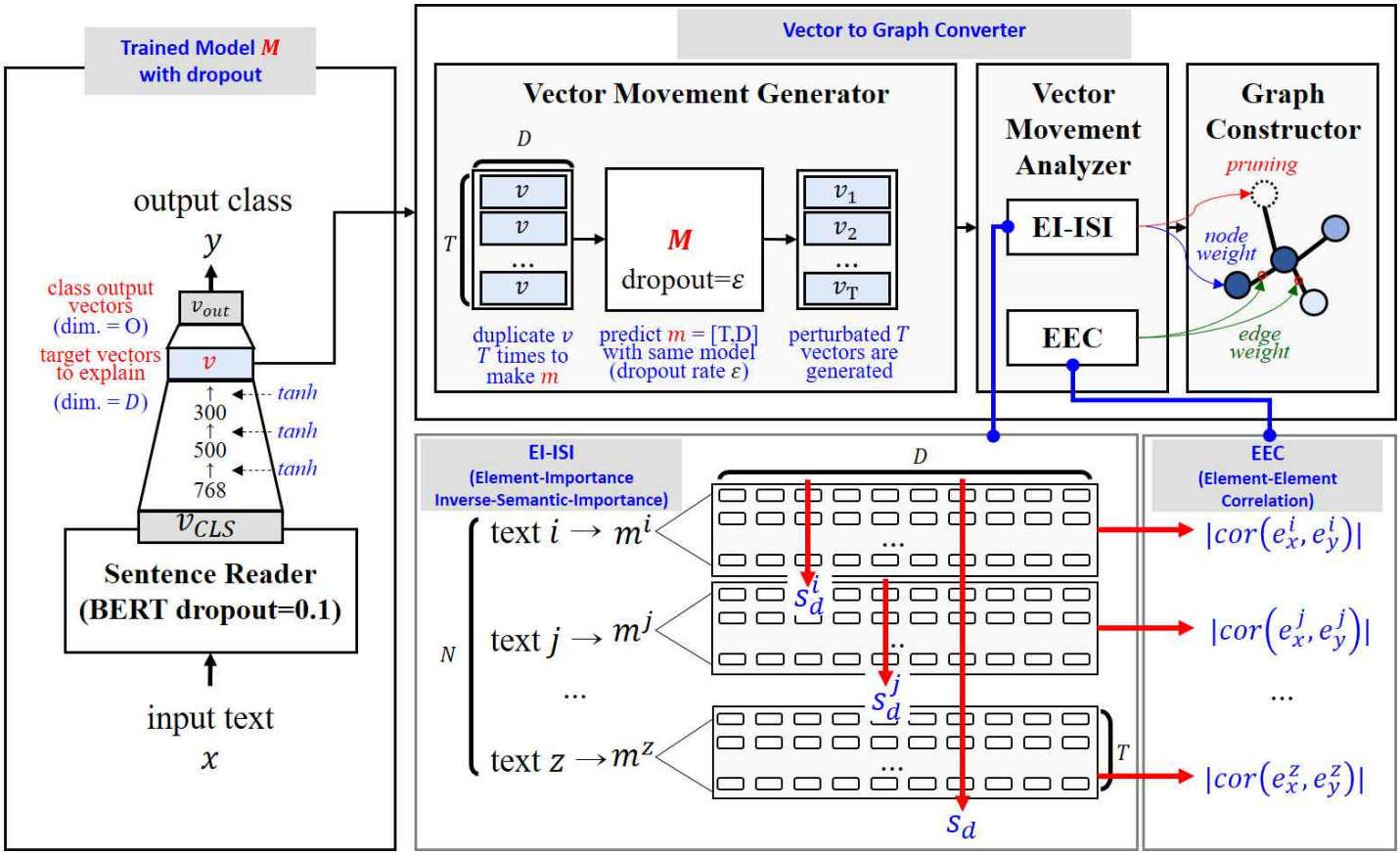

그림 2 벡터-그래프 변환 프레임워크의 전반적인 구조

본 연구에서는, 인간과 기계가 모두 이해 가능한 구조의 하나로 노드(Node)와 엣지(Edge)로 구성되는 그래프가 가장 표현력이 높고 전-후처리가 용이한 표현 형태임을 가정하고, 심층신경망의 내부 벡터가 그래프로 변환되는 방법론을 연구하였다. 인간이 그래프를 시각화하고 분석하는 것은 가공되지 않은 벡터(Raw Vector) 형태의 표현을 다루는 것보다 상대적으로 쉽다. 또한 그래프 구조를 처리하는 많은 알고리즘들이 컴퓨터 과학과 수학 분야에서 개발되었기에 그래프 구조는 기계에 의해 쉽게 처리될 수 있으며, 더 나아가 그래프 기반 분석 방법은 블랙박스(Black-box) 신경망의 내부 동작을 설명하는 새로운 가능성을 열 수도 있을 것이다.

본 논문에서는 딥러닝 기반 모델의 설명력을 제공하기 위해 벡터를 그래프로 간결하게 변환할 수 있는 **벡터-그래프 변환 프레임워크(Vector2graph Framework)**를 제안한다 (그림 1). 제안하는 프레임워크를 위해 모델 예측 단계에서 목표 벡터의 동적 움직임을 조사하기 위한 효과적인 섭동(Perturbation) 기법으로 드랍아웃(Dropout) [10]을 검토한다. 동적으로 섭동하는 움직임은 그래프 표현을 구성하기 위한 새로운 정보를 제공한다.

본 연구에서는 그래프의 노드와 엣지를 구성하기 위해 벡터 차원 중요도를 측정하는 EI-ISI(Element-Importance Inverse-Semantic-Importance) 점수를 제안하고 모든 벡터들 사이의 상호관계 정도를 나타내는 EEC(Element-Element-Correlation) 점수를 제안한다.

제안한 방법론의 타당성을 검증하기 위해 설계된 의도 분류(Intent Classification)[11] 실험 결과들은 제안한 변환 프레임워크가 벡터들로부터 핵심 구조를 추출하며 벡터를 그래프로 성공적으로 변환함을 증명한다. 마지막으로 다양한 그래프 기반 시각화 방법을 소개한다.

2. 관련 연구

최근 몇 년간 딥러닝의 설명 가능성을 제공하기 위해 많은 연구들이 진행되었다. 이 연구들은 시스템이 무엇에 집중하고 어떻게 결정을 내리는지를 인간이 이해할 수 있도록 도와주는 모달리티의 종류에 의해 분류될 수 있다.

몇몇 연구들은 예측 결과들을 설명할 수 있는 텍스트를 생성한다[4, 5]. [4]는 텍스트 기반 설명 데이터 셋을 구축하고 테스트 기간 동안 설명을 통합하는 모델을 구현했다. [5]는 인간이 읽을 수 있는 설명을 얻기 위한 방법으로 사용자가 작성한 평가 및 세분화된 요약을 제공하는 웹사이트를 통해 데이터 셋을 구축했다. 몇몇 다

른 연구들은 내부 결정 또는 결과들의 증거를 강조하기 위해 시각적 마커들을 사용했다[6, 7, 8]. [6]은 이미지 패치를 사용하여 해석 가능한 표현을 넘어서 해석 가능한 모델을 식별하려는 시도를 했다. [7]은 히트맵을 사용하여 시스템이 무엇에 집중하고 있는지 이해할 수 있도록 도왔다. [8]은 Constituency Parsing Tree를 설명하기 위해 긍정/부정 마커들을 도입했다.

이전 방법들과 달리 설명 가능성을 제공하기 위해 본 연구에서는 그래프 모달리티를 도입한다. 이와 관련하여, 직접적으로 벡터 임베딩을 그래프로 변환하려는 시도가 있었다. [9]는 이미지 처리 중에 합성곱 신경망(CNN)에 의해 생성되는 임베딩의 그래프 표현을 얻기 위한 방법론을 제안했다. 이와 달리, 본 논문에서는 드랍아웃을 사용하여 벡터의 동적 움직임을 생성하고 그래프를 구성한다.

3. 벡터-그래프 변환 프레임워크(Vector2graph Framework)

본 논문에서 제안하는 벡터-그래프 변환 프레임워크는 학습이 완료된 모델의 학습 파라미터에 대한 간섭 없이 벡터들의 동적 움직임을 분석함으로써 예측 단계에서 임베딩 벡터를 그래프로 변환한다. 이를 위해서 우리는 다음의 세 가지 문제를 고려하였다.

- 입력과 일치하는 목표 벡터의 동적 움직임을 생성하는 방법
- 그래프의 노드와 엣지를 구성하기 위한 정보 추출의 수단으로써 벡터의 동적 움직임을 분석하는 방법
- 벡터 표현법의 의미정보를 유지하면서 그래프로 변환하는 방법

위의 문제들을 해결하기 위해 제안한 프레임워크는 벡터의 동적 움직임 생성기, 벡터의 동적 움직임 분석기, 그래프 변환기의 세 가지 컴포넌트(Component)로 구성된다.

3.1 벡터의 동적 움직임 생성기

[10]에서 제안한 드랍아웃은 신경망 훈련 중에 과대적합을 피하기 위해 확률적으로 일부 뉴런의 출력을 생략하는 기법이다. 드랍아웃은 신경망에 무작위성(Randomness)을 제공함으로써 학습의 안정성과 정확성을 향상시킨다.

본 논문에서는 드랍아웃을 효과적인 섭동 기법의 관점에서 검토한다. 성공적으로 훈련되는 신경망은 드랍아웃의 무작위성을 통해 벡터 표현을 생성함으로써 훈련 단계에서 목적 함수에 맞는 입력 표현을 생성한다. 이는 특정 벡터 요소(Vector Element)가 무작위 섭동에도 살아남아 주어진 입력을 일관되게 표현한다는 것을 의미하고 이러한 요소들이 다른 요소들에 비해 안정적이어야만 한다는 것을 의미한다.

이러한 가정에 근거하여 본 연구에서는 표현력이 높은 벡터 요소들을 추출하기 위한 방법으로 예측 단계에서 드랍아웃을 사용한다. 벡터의 동적 움직임을 생성하는 과정은 다음과 같다.

1) 신경망 학습. 신경망은 드랍아웃이 적용되어 학습된다. 본 연구에서 사용된 신경망의 구조는 그림 2와 같다.

2) 변환할 벡터 선정. 변환될 벡터는 드랍아웃 층보다 출력 층에 더 가까워야 한다. 본 연구에서는 평면화 된 클래스별 벡터(Flattened Class-wise Vector) 아래층의 출력 벡터를 변환할 벡터로 선택하였다 (그림 2에서의 target vectors).

3) 섭동 기법을 통한 다양한 벡터 임베딩 생성. 전형적인 예측 단계에서는 드랍아웃이 적용되지 않기 때문에 모델의 출력은 항상 동일하다. 하지만 본 연구에서는 예측 단계에서도 드랍아웃을 적용하여 동일한 입력에 대해 서로 다른 벡터 표현을 생성한다.

3.2 벡터의 동적 움직임 분석기

동일한 입력에 대해서 섭동으로 인해 생성된 서로 다른 벡터 표현들 중에서 가장 대표적이고 중요한 정보를 포함하는 벡터 요소는 다른 요소보다 더 안정적이다. 따라서 본 연구에서는 표준편차(Standard Deviation)를 기반으로 안정성을 측정하는 EI-ISI(Element-Importance Inverse-Semantic-Importance) 점수를 제안한다.

같은 입력 i에 대해 드랍아웃을 통해 생성된 T개의 서로 다른 벡터들은 $V_t^i = [e_{1,t}^i, e_{2,t}^i, ..., e_{D,t}^i]$와 같이 표현할 수 있다. 이 때, $e_{d,t}^i$는 전체 차원의 크기가 D인 벡터들 중에서 t번째로 생성된 벡터의 d번째 요소이다. 우리는 다음과 같이 표본 표준편차 s_d^i를 통해 동일한 입력에 대한 각 요소의 동적 움직임을 측정한다.

$$\overline{X}_d^i = \frac{\sum_{t=1}^{T} e_{d,t}^i}{T}$$

$$s_d^i = \sqrt{\frac{\sum_{t=1}^{T} (e_{d,t}^i - \overline{X}_d^i)^2}{T-1}}$$

동일한 입력에 대해 중요한 요소는 안정적일 것이라는 가정에 근거하여 위의 수식에서 s_d^i값이 크다면 해당 입력에서 중요하지 않은 요소라는 의미를 갖는다. 이를 측정하는 과정은 그림 2에 표시되어 있다.

표 1 의도분류 실험 결과

	Weather	Navigation	M2M-M	M2M-R	Multilingual-en	Multilingual-es	Multilingual-th
BERT (text)	0.995	0.996	0.900	0.883	0.989	0.918	0.890
Graph	**0.993**	**0.992**	**0.897**	**0.874**	**0.990**	**0.950**	0.893
Graph w/o edges	0.990	0.990	0.895	**0.874**	**0.990**	0.947	**0.896**

유사하게 우리는 전체 입력에 대한 d번째 요소의 표준편차 s_d를 구할 수 있다. 같은 입력 i에 대한 d번째 요소의 표준편차 s_d^i와 모든 입력에 대한 d번째 요소의 표준편차 s_d를 비교하는 것은 중요한 통찰을 가져온다. s_d값이 작다면 다른 입력들에서도 중요한 요소일 수 있기 때문에 해당 입력 i에서는 중요한 요소가 아닐 수 있다. 따라서 우리는 입력 i에 대한 d번째 요소의 중요도를 측정하기 위해 **Element-Importance Inverse-Semantic-Importance(EI-ISI)** 점수를 다음과 같이 정의한다.

$$E_d^i = \frac{1/s_d^i}{1/s_d} = \frac{s_d}{s_d^i}$$

본 연구에서는 벡터의 특정 차원 요소 1개가 변환된 그래프의 노드 1개에 해당한다고 가정한다. 예를 들어, 목표 벡터가 100차원이라면, 100개의 노드로 구성된 그래프가 만들어 진다.

그래프를 구성하기 위해서는 각 노드간의 관계를 표현하는 엣지 정보가 필요하다. 이를 위해 우리는 생성된 벡터들 각 요소의 상관관계를 사용하는 **Element-Element Correlation(EEC)** 점수를 정의한다. 특정 입력을 표현하는데 사용되는 표현력이 높은 벡터 요소들은 동시에 움직이므로 핵심 요소간의 움직임이 서로 연관되어야 한다는 것이 EEC 점수를 고안하게 만든 배경이다. 예를 들어, 바람에 흔들리는 나무를 상상할 때, 나무 외곽의 가지들과 잎들의 움직임에는 일관성이 없을 것이다. 그러나, 나무의 기둥에 가까운 주요가지들은 바람이 불 때 같은 방향으로 흔들릴 것이다. 섭동 기법에 의해 생성된 벡터의 동적 움직임도 이와 유사할 것이라고 생각할 수 있다. EEC 점수는 다음과 같이 정의된다.

$$EEC_{x,y}^i = |cor(e_x^i, e_y^i)|$$

위의 수식에서 e_x^i는 입력 i에 대한 표본들($e_{x,t}^i$)을 의미하고 cor은 피어슨 상관계수를 의미한다. 피어슨 상관계수는 생물학적 네트워크 분석에서 부호가 없는(Unsigned) 유전자 공동 발현 유사성(Gene Co-Expression Similarity)으로도 알려져 있다[12]. 생물학에서 유전자 공동 발현 유사성은 유전자들 사이의 엣지 정보로 사용된다. 이는 우리의 방법론과 같다.

표 2 의도분류 실험에 사용된 데이터 셋 통계

	# of trains	# of devs	# of tests	# of intents
Weather	6,993	3,009	2,998	14
Navigation	5,601	-	2,402	8
M2M-M	3,321	1,033	2,291	29
M2M-R	10,653	2,443	5,707	36
Multilingual-en	30,521	4,181	8,621	12
Multilingual-es	3,617	1,983	3,043	12
Multilingual-th	2,156	1,235	1,692	12

3.3 그래프 변환기

본 연구에서 제안하는 그래프 변환기는 목표 벡터의 각 요소를 그래프의 노드와 일치한다고 가정하고 각 요소의 중요도를 나타내는 EI-ISI 점수 값을 할당한다. 이때, 그래프의 노드와 벡터 요소가 일치하기 때문에 EI-ISI 점수는 그래프에서 각 노드의 중요도를 나타나게 된다. 또한 각 노드의 관계 정도를 나타내는 EEC 점수 값은 각 노드간의 가중 엣지(Weighted Edge)로 사용된다.

그래프를 구성하기 위해서 모든 노드들을 사용하는 것은 그래프가 너무 복잡해질 뿐만 아니라 인간과 기계가 이해할 수 있는 표현법을 추구하는 본 연구의 목적과 부합하지 않는다. 따라서 본 연구에서는 EI-ISI 점수가 가장 높은 10개의 노드만 사용했다.

EEC 점수는 각 노드간의 연관성 정도를 의미하므로 우리는 이 점수를 직접적으로 그래프의 가중 엣지에 할당하였다. 또한 본 논문에서는 EEC 점수가 0.8보다 클 때 노드 간 엣지를 구성하도록 하였다.

4. 실험

제안한 방법론의 타당성을 검증하기 위해, 자연어 이해 문제 중 하나인 의도분류 말뭉치를 활용한 실험을 진행하였다. 추가적으로 그래프 기반의 새로운 시각화 방법을 소개한다.

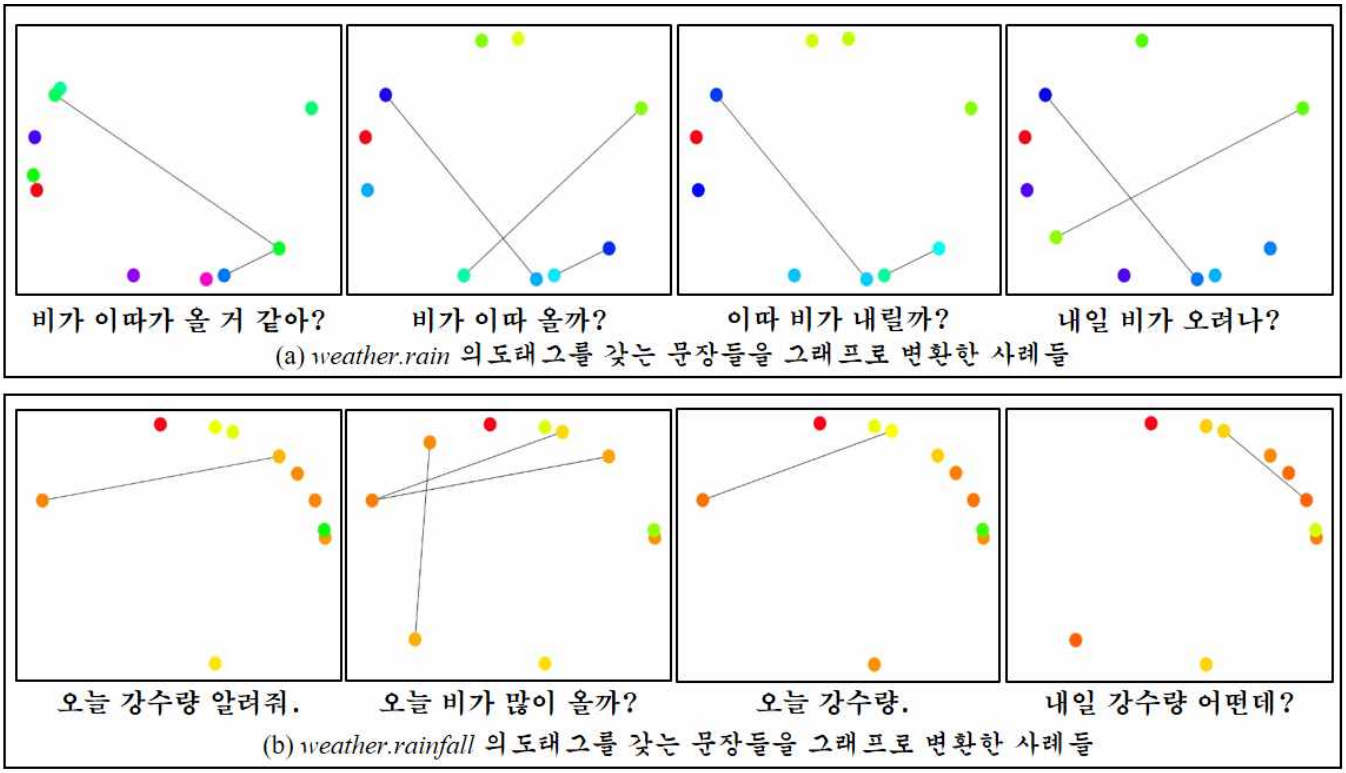

그림 3 의도태그에 따른 변환된 그래프 비교

4.1 의도분류 실험

본 연구의 목적은 벡터 표현법을 인간과 기계가 모두 이해할 수 있는 그래프 표현법으로 변환하는 것이다. 변환된 그래프는 벡터에 담겨있는 의미적 정보를 유지해야 한다. 본 논문에서는 이를 검증하기 위해서 의도분류 실험을 진행했다. 한국어 데이터 셋인 Weather, Navigation 데이터 셋과 영어 데이터 셋인 M2M-M(Movie), M2M-R(Restaurant)[13]과 영어, 스페인어, 태국어를 포함하는 데이터 셋인 Multilingual Task Oriented Dataset[14]을 이용했다. M2M-M, R 데이터 셋들은 3-4개의 대화 내용을 포함하고 있다. 그러나 본 연구에서 사용하는 신경망은 문장 단위의 분류기이다. 따라서 M2M-M, R 데이터 셋을 문장 단위로 구분하여 사용하였다. 전체 데이터 셋의 통계는 표 2와 같다.

실험에 사용된 모든 의도분류기들은 드랍아웃 디폴트 비율이 0.1인 BERT sentence reader[1]를 사용하였다. 그림 2는 의도분류 실험에 사용된 신경망의 구조를 보여준다. 또한 드랍아웃을 통해 생성되는 목표 벡터의 수는 30개로 고정하였다.

본 논문에서 제안하는 벡터-그래프 변환 프레임워크를 사용하면 모든 훈련 데이터들과 테스트 데이터들은 각각 훈련 그래프들($G_{train}s$)과 테스트 그래프들($G_{test}s$)로 변환된다. 의도태그들을 구분할 수 있는 충분한 의미적 정보를 변환된 그래프들이 담고 있다면 그래프 이미지들을 사용하여 모델을 학습하고 의도태그들을 분류할 수 있어야 한다. 이를 검증하기 위해 [15]의 image reader 네트워크를 사용하여 전형적인 CNN기반 이미지 분류 훈련 및 예측 실험을 수행한다. 다시 말해 CNN 기반의 이미지 분류기들은 $G_{train}s$의 이미지들을 사용해서 학습되고 $G_{train}s$의 이미지들을 통해 평가되어 성공적으로 벡터에서 그래프로의 변환이 이루어졌다면 텍스트 기반의 sentence reader와 실험 성능이 유사할 것이라고 생각할 수 있다.

표 1은 BERT sentence reader를 통한 의도분류 실험과 변환된 그래프 이미지들을 통한 의도태그에 대한 이미지 분류 실험의 결과를 보여준다. 표 1의 실험 결과를 통해 성공적으로 제안하는 프레임워크가 벡터를 그래프로 변환함을 확인할 수 있다. 또한 Multilingual 데이터 셋에 대해서는 CNN 기반의 이미지 분류기의 성능이 더 우수하다. 이는 그래프로 데이터를 표현하는 것이 모델의 성능의 향상에도 도움이 될 수 있음을 암시한다.

4.2 그래프 기반 표현법 시각화

본 연구의 주요 기여 중 하나는 심층신경망의 내적인 상태를 시각화하는 새로운 방법을 제안하는 것이다. 그림 3은 weather 말뭉치에서 실제로 변환된 그래프들을 의도태그에 따라 구분하여 보여준다. 이러한 시각화 방법을 통해, 벡터의 숫자들을 살펴보는 것보다 더 명확하게 의도태그에 따른 차이를 파악할 수 있다. 그림 3을 통해 확인할 수 있듯이 변환된 그래프들은 의도태그에 따라 상당히 다름을 확인할 수 있다. 또한 서로 같은 의도를 갖는 문장들은 다소 그래프의 형태가 유사함을 확

인할 수 있다.

5. 결론

본 논문에서는 신경망의 내적인 동작 또는 예측 결과를 설명하기 위해 그래프를 유용한 모달리티로 제안하고 이를 위해 벡터를 그래프로 변환하는 새로운 프레임워크를 제안했다. 벡터-그래프 변환 프레임워크는 드랍아웃을 사용하는 기존의 훈련된 네트워크에 연결될 수 있다. 제안한 프레임워크에서 드랍아웃 기술은 같은 입력에 대한 벡터의 움직임을 생성하기 위해서 적용되고 프레임워크 내의 움직임 분석기는 유용한 벡터 요소와 벡터 요소 간의 관계를 추출하기 위해 사용된다. 이 때, 유용한 벡터 요소는 본 연구에서 제안한 Element-Importance Inverse-Semantics-Importance 점수를 통해 측정되고 벡터 요소 간의 관계는 Element-Element-Correlation 점수를 통해 측정된다. 제안한 점수들은 그래프 구조를 구성하는데 사용된다. Weather, Navigation, M2M-M/R, Multilingual 데이터 셋을 통해 본 논문에서는 제안한 프레임워크가 의미론적 정보를 유지하면서 벡터 표현을 그래프 표현으로 성공적으로 변환함을 보여주었다. 추가적으로 내적인 의미 표현법들과 그들의 관계를 그래프 기반 표현법을 통해 시각화하는 방법을 보여주었다.

제안한 연구 결과에 기반 하여 향후에는 다양한 연구 방향들이 고려될 수 있다. 예를 들어, 벡터-그래프 변환 프레임워크는 기계번역과 같은 더 복잡한 자연어 처리 문제들에 이용될 수 있다. 또한 벡터-그래프 변환 프레임워크는 컴퓨터 비전, 음성과 같은 다른 분야에도 적용될 수 있으며 이에 대한 향후 연구들이 진행될 수 있을 것으로 기대한다.

감사의 글

이 논문은 2020년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임(No.2020-0-01441)

이 논문은 2019년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임(2019-0-00004, 준지도학습형 언어지능 원천기술 및 이에 기반한 외국인 지원용 한국어 튜터링 서비스 개발)

이 논문은 2019년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(2019R1F1A1060601)

참고문헌

[1] Jascob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, 2019.

[2] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut, “Albert: A lite bert for self-supervised learning of language representations,” In Proceedings of the 2020 Conference of the International Conference on Learning Representations, 2020.

[3] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," arXiv preprint arXiv:1910.10683, 2019.

[4] Oana-Maria Camburu, Tim Rockt¨aschel, Thomas Lukasiewicz, and Phil Blunsom, "e-snli: Natural language inference with natural language explanations," In Advances in Neural Information Processing Systems, pages 9539–9549, 2018.

[5] Hui Liu, Qingyu Yin, and William Yang Wang, "Towards explainable nlp: A generative explanation framework for text classification," arXiv preprint arXiv:1811.00196, 2018.

[6] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin, " why should i trust you?" explaining the predictions of any classifier, In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, pages 1135–1144, 2016.

[7] Wojciech Samek, Thomas Wiegand, and Klaus-Robert M¨uller, "Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models," arXiv preprint arXiv:1708.08296, 2017.

[8] Xisen Jin, Junyi Du, Zhongyu Wei, Xiangyang Xue, and Xiang Ren, "Towards hierarchical importance attribution: Explaining compositional semantics for neural sequence models," arXiv preprint arXiv:1911.06194, 2019.

[9] Dario Garcia-Gasulla, Armand Vilalta, Ferran Par´es, Jonatan Moreno, Eduard Ayguad´e, Jes´us Labarta, Ulises Cort´es, and Toyotaro Suzumura, "Building graph representations of deep vector embeddings," arXiv preprint arXiv:1707.07465, 2017.

[10] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov, "Improving neural networks by preventing coadaptation of feature detectors," arXiv preprint arXiv:1207.0580, 2012.

[11] Patti J Price, "Evaluation of Spoken Language Systems: the ATIS Domain," In Speech and Natural Language: Proceedings of a Workshop Held at Hidden Valley, Pennsylvania, June 24-27, 1990.

[12] Lin Song, Peter Langfelder, and Steve Horvath, "Comparison of co-expression measures: mutual information, correlation, and model based indices," BMC bioinformatics, 13(1):328, 2012.

[13] Pararth Shah, Dilek Hakkani-Tür, Gokhan Tür, Abhinav Rastogi, Ankur Bapna, Neha Nayak, Larry Heck, “Building a Conversational Agent Overnight with Dialogue Self-Play,” arXiv preprint arXiv:1801.04871, 2018.

[14] Sebastian Schuster and Sonal Gupta and Rushin Shah and Mike Lewis, “Cross-Lingual Transfer Learning for Multilingual Task Oriented Dialog,” In proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Shor Papers), pages 3795-3805, 2019.

[15] Yuntian Deng, Anssi Kanervisto, and Alexander M Rush, "What you get is what you see: A visual markup decompiler," arXiv preprint arXiv:1609.04938, 10:32–37, 2016.