



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

박사학위논문

BERT 기반의 전이학습 모델을 적용한 양방향 인재매칭 시스템

Bidirectional HR Matching system
Using a BERT-based Transfer Learning Model

2022년 02월

한남대학교 대학원

경영정보학과

오 소 진



한남대학교
Hannam University

BERT 기반의 전이학습 모델을 적용한 양방향 인재매칭 시스템

Bidirectional HR Matching system
Using a BERT-based Transfer Learning Model

지도교수 송희석

이 논문을 박사학위 논문으로 제출함

2022년 02월

한남대학교 대학원

경영정보학과

오 소 진



한남대학교
Hannam University

오 소 진의 박사 학위논문으로 확인함

심사위원장 박 광 일 ㉠

심 사 위 원 송 희 석 ㉠

심 사 위 원 김 재 경 ㉠

심 사 위 원 장 문 경 ㉠

심 사 위 원 강 신 철 ㉠

2022년 2월

한남대학교 대학원



한남대학교
Hannam University

감사의 글

연구자로서의 첫발을 딛는 지금, 학부 졸업 후 사회에 첫발을 디뎠던 때와 비슷한 마음입니다. 두려움, 앞으로 해야 할 일에 대한 무게를 느끼면서도 너무나 벅차고 설레는 마음이 더 큼니다. 그동안 저를 위해 애써 주시고 도와주신 분들이 너무 많아 어떻게 마음을 모두 표현해야 할지 모를 지경입니다. 감사의 글로나마 그 마음을 전하고자 합니다.

학부 졸업 후 10년 동안 개발자로 일을 하면서, 여러 가지 사정들로 인해 슬럼프에 빠져 있을 때, 송희석 교수님께 저의 힘든 마음과 일에 대한 고민을 담아 메일 한 통을 보냈었습니다. 연구년이셔서 해외 계셨음에도 불구하고 바로 답장을 주시며 선뜻 저를 제자로 받아주셨습니다. 학부에서 배운 것으로 10년을 일 했으니, 다음 10년을 위해 다시 공부하면 된다고 하며 육아를 병행할 수 있도록 배려해 주셨습니다. 송희석 교수님께 학부 시절 배운 프로그래밍을 통해 개발자로 살았고, 학위과정을 통해 딥러닝 연구자로 다시 한번 도약할 수 있게 해 주신 저의 지도 교수님이시자, 제 인생의 최고의 스승이시고 저를 지금 이 자리에 있게 해 주신 분입니다. 너무 감사드리고 앞으로도 발전하는 모습 보여드리겠습니다.

제가 심적으로 힘들 때마다 항상 위로해 주시고, 항상 잘 하고 있다고 지지해 주시고, 제가 하고 싶은 일을 모두 할 수 있도록 학과에서 전폭적인 지원을 아끼지 않으셨던 김재경 교수님 정말 감사드립니다. 감사한 마음을 보답할 수 있도록 지금처럼 같은 마음으로 열심히 노력하겠습니다.



학위과정은 연구의 길을 만들어가는 과정이라고 생각하는데, 그 길을 만들어 갈 때 저의 페이스메이커가 되어 주신 장문경 교수님 정말 감사드립니다. 교수님께서 옆에서 달려주시지 않았다면 완주하지 못했을 겁니다. 항상 제가 방향을 잃거나 주춤할 때 먼저 손을 내밀어 주시고 다시 달릴 수 있도록 밀어주셨습니다. 다시 한번 감사드립니다.

더불어 학부 시절부터 지도해 주셨던 남수현 교수님, 강신철 교수님, 박광일 교수님 감사드립니다. 20년 전 꼬맹이가 이제 박사가 되었습니다. 모두 교수님께서 지도해 주시고 관심 있게 봐주셨기 때문에 가능한 일입니다. 다시 한번 감사드립니다.

또한, 학위과정에서 같이 수업도 듣고 연구도 했던 이제는 모두 박사님이 되신 김병엽 박사님, 박상호 박사님 소중한 추억으로 간직하겠습니다. 김병엽 박사님, 저의 박사 학위 준비 많이 도와주셔서 감사합니다.

저의 인생의 멘토이신 채은경 대표님, 육아와 일을 병행하며 힘들어할 때 먼저 경험하신 것들 공유해 주시고, 아이 교육에 대한 좋은 정보도 알려주시고 정말 저에게는 인생 선배이자 멘토이십니다. 항상 감사드리고 앞으로도 계속 보답하며 같이 할 수 있길 바랍니다.

잘 다니던 회사를 그만두고 학위를 한다고 했을 때, 지지해 주고 항상 옆에서 지켜봐 준 나의 든든한 남편 박진호님께 진심으로 감사드립니다. 공부한다고 아내 역할 한번 한 적 없는데 항상 이해해줘서 정말 고맙고 또 고맙고 이 마음 잊지 않고 앞으로 더 잘하도록 하겠습니다. 또한, 우리 가족들 시아버지, 시어머니, 아빠, 엄마, 언니, 남동생, 형부, 도련님,

동서까지 항상 저를 지지해 주셔서 감사드리고, 특히 아이 봐주시느라고
고생해주신 엄마, 아빠 정말 감사드려요. 효도하겠습니다.

마지막으로 제가 학위과정을 해야겠다 마음먹게 해 준 사랑하는 나의
딸, 나의 소울메이트 박지원. 바쁘고 힘들어할 때 어린테도 엄마 이해해
주고 오히려 힘내라고 해 줘서 고마워. 엄마가 지원이 엄마로서 멋진 사
람으로 살 수 있도록 계속 노력할게. 사랑한다.

감사한 분들 너무 많은데 지면에 다 하지 못했지만, 포기하지 않고 논문
을 마무리할 수 있도록 격려와 지지를 보내주신 모든 분께 이 자리를 빌
려 거듭 감사드립니다.

이 첫걸음의 마음을 기억하며, 항상 배우는 자세와 겸손함을 가지고 새
로운 연구를 도전하겠습니다. 또한, 제가 가진 지식을 나눠주고 공유하며
베풀면서 살 수 있도록 하겠습니다.

2022년 2월

오 소 진



한남대학교
Hannam University

목 차

I. 서 론	1
1. 연구의 배경	1
2. 연구의 목적	3
3. 연구의 구성	6
II. 이론적 배경	7
1. 일자리 추천 시스템	7
1) 추천 시스템	7
2) 일자리 추천 시스템	15
2. 자연어 처리	23
1) 자연어 처리	23
2) 전이 학습(Transfer Learning)	28
3) 어텐션 메커니즘(Attention mechanism)	34
4) 트랜스포머(Transformer)	39
5) BERT	42
III. 양방향 인재매칭 모델	52
1. BERT 기반의 전이학습 모델 설계	52
1) BERT 모델 입력	55

2) BERT 모델 미세조정	58
3) BERT 모델 하이퍼 파라미터 튜닝	59
IV. 실험 및 분석결과	62
1. 데이터 셋	62
2. 성능평가 방법	63
3. 실험 결과	67
V. 양방향 인재매칭 시스템 제안	74
1. 양방향 인재매칭 시스템 설계	74
1) 양방향 인재매칭 시스템 구성	74
2) 양방향 인재매칭 시스템 서비스 프로세스	75
3) 양방향 인재매칭 시스템 API 설계	77
4) 양방향 인재매칭 시스템 예상 결과물	81
VI. 결 론	85
1. 연구의 요약 및 공헌	85
2. 연구의 한계 및 향후 연구	88
참 고 문 헌	90
영 문 초 록	105

표 목 차

<표 II-1> 사용자-아이템 행렬 예시	10
<표 II-2> 추천 시스템의 추천 기법	13
<표 II-3> 추천 시스템의 장·단점	13
<표 II-4> 일자리 추천 시스템 비교	18
<표 II-5> AI 기반 구직 매칭 솔루션	21
<표 II-6> 통계적 기반 임베딩 기법	26
<표 II-7> 신경 네트워크 기반 임베딩 기법	27
<표 II-8> 심층 전이학습의 분류	32
<표 II-9> BERT를 이용한 자연어 처리 분야 주요 연구	50
<표 III-1> 하이퍼 파라미터 튜닝 기법	59
<표 III-2> BERT 미세조정 및 하이퍼 파라미터	61
<표 IV-1> 실험 데이터 셋	62
<표 IV-2> 정밀도 계산 공식	64
<표 IV-3> 재현율 계산 공식	64
<표 IV-4> f1-score 계산 공식	64
<표 IV-5> Top-N 추천 재현율	65
<표 IV-6> Top-N 추천 정확도	66
<표 IV-7> 정밀도, 재현율, f1-score, AUC 모델 성능 비교	68
<표 IV-8> Top-N 추천 재현율과 Top-N 추천 정확도 추천 성 능 비교	72

<표 V-1> 양방향 인재매칭 Web API	77
<표 V-2> topNHrs API 사용 방법	78
<표 V-3> topNHrs API 요청 결과 JSON 포맷	79
<표 V-4> topNHrs API statusCode 정의표	79
<표 V-5> topNJobs API 사용 방법	80
<표 V-6> topNJobs API 요청 결과 JSON 포맷	81
<표 V-7> topNJobs API statusCode 정의표	81

그림 목차

<그림 II-1> 추천 시스템의 분류	8
<그림 II-2> 사용자 기반의 협업 필터링 기법	9
<그림 II-3> 아이템 기반 협업 필터링 기법	10
<그림 II-4> 내용 기반 필터링 추천 기법	12
<그림 II-5> 최근 6년 간 ACL, EMNLP, EACL, NAACL에 게재 된 딥러닝 논문의 비율(%)	24
<그림 II-6> 임베딩(Embedding) 예시	25
<그림 II-7> 기계학습 과정과 전이학습 과정 비교	29
<그림 II-8> 전이학습 과정	31
<그림 II-9> 기본적인 seq2seq 모델	35
<그림 II-10> LSTM seq2seq 모델	36
<그림 II-11> 어텐션 메커니즘의 예시	37
<그림 II-12> 글로벌 어텐션과 로컬 어텐션 차이	38
<그림 II-13> 트랜스포머 모델 구조	40
<그림 II-14> 상호 참조 문제 프랑스어 번역 구문 예시	41
<그림 II-15> 훈련된 트랜스포머의 인코더 셀프 어텐션 분포	41
<그림 II-16> BERT 모델 학습	43
<그림 II-17> BERT 데이터 입력 표현	44
<그림 II-18> 다양한 작업(Different Tasks)에 대한 BERT 미세 조정 방법	46
<그림 III-1> 내용 기반 척도학습모형	53

<그림 III-2> BERT 기반 양방향 인재매칭 모델	54
<그림 III-3> BERT 기반 양방향 인재매칭 모델 입력	55
<그림 III-4> 구직자 프로파일	56
<그림 III-5> 채용정보 프로파일	56
<그림 III-6> 지원 이력 데이터	57
<그림 III-7> 통합 데이터	57
<그림 III-8> BERT 기반 양방향 인재매칭 모델 미세조정	58
<그림 IV-1> 정밀도, 재현율, f1-score, AUC 성능 비교	70
<그림 IV-2> ROC 곡선 그래프 성능 비교	71
<그림 IV-3> Top-N 추천 재현율(REC@N) 성능 비교	73
<그림 IV-4> Top-N 추천 정확도(AAS@N) 성능 비교	73
<그림 V-1> 양방향 인재매칭 시스템 구성도	75
<그림 V-2> 양방향 인재매칭 시스템 기능 흐름도	76
<그림 V-3> 기업 채용정보 목록 UI	82
<그림 V-4> 기업 사용자 인재매칭 UI	83
<그림 V-5> 구직자 e-포트폴리오 UI	83
<그림 V-6> 구직자 채용정보 매칭 UI	84

I. 서 론

1. 연구의 배경

최근 ‘코로나 19세대’ 라는 말이 신조어로 등장할 정도로 코로나 19의 장기화는 계속되어 세계 경제뿐만 아니라 우리나라 경제에 미치는 피해도 심각한 수준에 직면해 있다. 통계청이 발표한 2021년 2월 기준 자료에 따르면 실제 청년실업률은 10.1% 두 자릿수를 기록 하였고, 청년 실업자는 41만 6천 명에 이른다(통계청, 2021). 그중 우리 경제의 허리에 해당하는 30대 취업자 감소세가 이어지는 모습이다.

한편 정부는 청년 취업난을 해소하기 위해 마련한 ‘청년 지원금’ , ‘청년 대출’ , ‘취업 성공패키지’ 등이 추진되고, 2021년 8월 기재부 등 4개 주요부처에 ‘청년’ 전담부서가 신설될 정도로 청년 취업난은 매우 심각한 상황이다(행정안전부, 2021).

이러한 상황에서도 기업은 여전히 구인난을 겪고 있다. 최근 고용 흐름의 3가지 특징 중 하나로 현장에서의 구인난은 커지는데, 좋은 일자리를 찾아 자발적으로 취업을 연기하는 구직자(취업준비생)는 늘어나는 미스매치 현상이 심화되고 있다(한국경영자총협회, 2021). 한국경영자총협회(2021) 3분기 보고서에 따르면 2021년 상반기 인력 부족률은 2.2%, 인력 부족 인원은 28.2만 명에 달하면서 전년 동기보다 인력 부족률은 0.4% 상승하고 인력 부족 인원은 5.9만 명 증가(상용근로자 5인 이상 사업체 기준)하여, 기업의 인력 부족률은 2018년 하반기(2.2%) 이후 최고

치, 부족 인원은 2018년 상반기(29.6만 명) 이후 최고치를 기록했다.

기업의 구인난과 청년 취업난의 가장 근본적인 원인은 일자리 미스매치(Job-Mismatch) 현상 때문이다(손배원, 2015). 일자리 미스매치 현상은 청년들이 가지고 있는 직무역량(Job competency)과 기업들이 현장에서 실제로 원하는 직무역량 간에 차이를 말한다. 이러한 일자리 미스매치 현상을 제대로 해결되지 않으면, 우리나라 청년실업률이 지금보다 낮아질 가능성은 그리 높지 않다. KDI 한국개발연구원에 따르면 2014~2017년 평균 청년실업률 3.62 % 중 일자리 미스매치 실업률(93%)이 노동력 부족 실업률 (7%)보다 절대적으로 높게 나타났다(김지운, 2018).

일자리 미스매치를 해소하기 위해 기업에는 기업의 채용정보에서 요구하는 역량과 기술을 보유한 인재를 추천하고, 구직자에게는 구직자가 보유한 역량과 기술에 적합한 일자리를 추천하는 양방향 인재 및 일자리 추천을 포함하는 개념으로 본 연구에서는 인재매칭으로 정의한다.

위와 같이 양방향 인재매칭 시스템을 통해 일자리 미스매치 해소를 위한 방법에 대해 관심을 가지게 되었으며, 일자리 미스매치 해소를 위하여 기업의 채용정보와 구직자의 이력서의 내용의 연관성을 이해하여 추천할 수 있는 전이학습 모델을 적용한 양방향 인재매칭 모델과 이를 실제 시스템에 적용할 수 있는 시스템 설계를 제안하고자 한다.

2. 연구의 목적

효과적인 일자리 추천을 위해서는 기업의 요구사항이 담겨 있는 채용 정보와 구직자의 역량과 기술이 담겨 있는 이력서를 추천에 활용하는 내용 기반 추천이 바람직하다고 알려져 있다(송희석, 2020).

기존 내용 기반 일자리 추천 연구에서는 구직자를 위한 일자리 추천과 기업을 위한 인재추천 각각 진행됐고, 송희석(2020)의 연구에서는 양방향 추천을 위해 내용 기반 척도학습 모델(Content-based metric learning model)을 제안한 바 있는데 이 방식은 기업의 취업공고와 구직자의 이력서에서 수집된 단어들의 출현 빈도(Frequency)에 집중하여 단어 가방(BoW: bag-of-words)으로 만들고 학습을 진행하고 있다. 단어 사전 구축은 자연어로 작성된 구직자의 이력서와 채용정보에 기술된 단어의 출현 빈도를 이용하여 단어의 유사성을 학습하기 때문에 문장에서 나타나는 단어와 단어 사이의 순서가 고려되지 않는 한계점이 있다. 또한, 단어 사이에 내재되어 있는 의미 즉, 같은 의미이지만 다른 단어로 기술되어 있는 경우 숨은 의미와 관계를 파악하기 어렵다. 결정적으로 단어 사전에 구축되어 있지 않은 새로운 단어에 대해서는 유사성을 학습할 수 없어 주기적으로 단어 사전을 재학습 시켜야 하는 단점이 있다.

이러한 한계를 극복하기 위해 본 연구는 자연어 처리 전이학습 모델은 BERT를 활용한 양방향 인재매칭 모델을 제안한다. 제안한 모델을 기존 내용 기반 척도 모델과 성능을 비교함으로써 제안한 모델의 유용성을 확인하고자 한다.

전이학습 모델을 적용하여, 신규 채용정보 및 이력서에 대한 단어 사전 재학습 비용 및 단어 사전 구축을 위한 시간을 절약할 수 있을 뿐 아니라, 추천 시스템이 새로운 기업의 채용정보 또는 구직자에 대한 충분한 정보가 수집된 상태가 아니라서 해당 구직자와 기업에 적절한 인재와 일 자리를 추천해주지 못하는 문제인 콜드 스타트(Cold start) (Bobadilla Sancho et al., 2012)를 해결하여 실제 추천 시스템으로 상용화 서비스를 제공할 수 있다.

청년층의 주된 취업 경로 중 공공·민간 직업알선기관, 취업박람회를 통한 취업은 12.5%에 불과(통계청, 2021)하며, 일자리 예산(33.6조 원) 중 직업훈련(7.1%), 고용서비스(3.8%)에 배정된 예산은 10% 수준에 불과하다(고용노동부, 2020). 취업을 위해서는 취업 포털 사이트 및 그 외 다른 경로를 통해 각자 살길을 찾고 있다. 현재 취업 포털 사이트에서는 AI 기반, 머신러닝 알고리즘을 적용한 일자리 추천 서비스를 진행하고 있지만, 실제 어떤 알고리즘이 적용되고 있는지 공개하지 않고 있다. 이에 본 연구에서 실험을 통해 검증된 양방향 인재매칭 모델을 적용한 인재매칭 시스템 구성 및 인재매칭 모델 API 연동 방법을 제시한다. 이에 실제 연구한 모델의 상용화 서비스를 위한 시스템 구성과 설계를 제공하고자 한다.

본 연구에서는 최근 기계 번역 분야 및 자연어 처리 분야에서 우수한 성능을 나타내고 있는 전이학습 모델인 BERT(Bidirectional Encoder Representations from Transformers)를 인재매칭 분야에 적용하여 양방향 인재매칭 모델을 설계하고, 제안한 양방향 인재매칭 모델을 적용한 인재매칭 시스템 구성을 제시함으로써 구직자에게는 구직을 위한 채용 정

보 검색 비용을 낮추고, 기업에는 기업이 원하는 인재를 추천하여 헤드헌터 고용 및 채용 광고 등과 같은 채용을 위한 추가적인 비용을 줄일 수 있으며, 나아가 구직자와 기업 모두에게 일자리 미스매치를 해소할 수 있는 서비스를 제공할 수 있을 것이다.

3. 연구의 구성

제1장은 서론으로 본 연구를 진행하게 된 배경과 목적을 기술하였으며, 구체적인 연구의 구성을 서술하였다.

제2장은 이론적 배경에서는 일자리 추천 시스템, 자연어 처리와 전이학습, 어텐션 메커니즘(Attention mechanism), 트랜스포머(Transformer) 모델, BERT 모델 관련된 선행연구, 본 연구와 기존 연구와의 차이점 등을 정리하였다.

제3장은 이론적 배경을 근거로 BERT 기반의 전이학습 모델 설계하였다. BERT 기반의 모델을 적용하기 위한 인코딩, 미세조정, 하이퍼 파라미터 튜닝을 설명하였다.

제4장은 제안한 모델의 성능을 평가하기 위한 실험용 데이터 셋을 설명하고, 성능평가 방법을 정의하였다. 또한, 기존 양방향 추천 내용 기반 척도학습 모델과 제안한 양방향 인재매칭 모델의 성능평가를 진행하여 연구 결과를 비교하였다.

제5장은 제안한 모델을 적용한 양방향 인재매칭 시스템을 제안하였다. 양방향 인재매칭 시스템의 구성도와 기능 흐름도, 추천 모델의 API 연동 방법 및 예상 결과물에 대하여 정리하였다.

제6장은 연구에 관한 결과를 요약하였으며, 연구의 시사점과 연구의 한계에 관하여 서술하였다. 그리고 향후 연구 방향에 대하여 제안하였다.

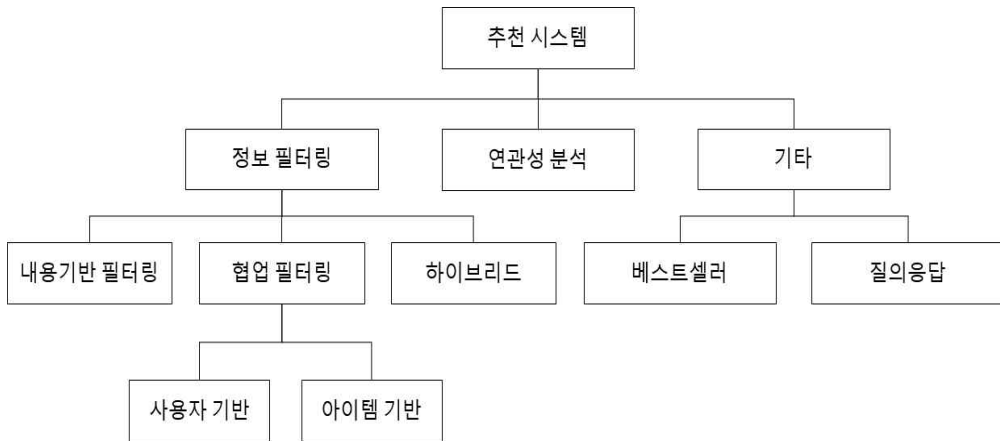
Ⅱ. 이론적 배경

1. 일자리 추천 시스템

1) 추천 시스템

추천 시스템(Recommender System)은 아이템의 선호도와 사용자의 과거 활동 이력에 의한 정보 필터링 기법을 활용하여 사용자에게 선호도 높은 아이템을 표출하는 시스템이며, 사용자 구매 및 매출 증가에 주로 활용된다(Dacrema, 2019). 사용자로서는 다양해지는 아이템 종류와 사전 지식의 부족으로 자신이 원하는 물건을 선택하는데 어려워지고, 판매자로서는 개개인의 선호도를 고려해 적절한 아이템을 추천해 이를 구매로 연결해 주기 때문에 적절한 아이템 추천이 중요하다(Das et al., 2013). 추천 시스템에 활용되는 필터링 기법 연구는 1990년대 시작되어, 2006년 10월에 Netflix사에서 개최한 Netflix Prize 경연대회(Bigdeli & Bahmani, 2008)를 기점으로 많은 연구와 관심이 집중되었다.

그림 <그림 Ⅱ-1>과 같이 다양한 추천 시스템의 종류를 보여주고 있다.

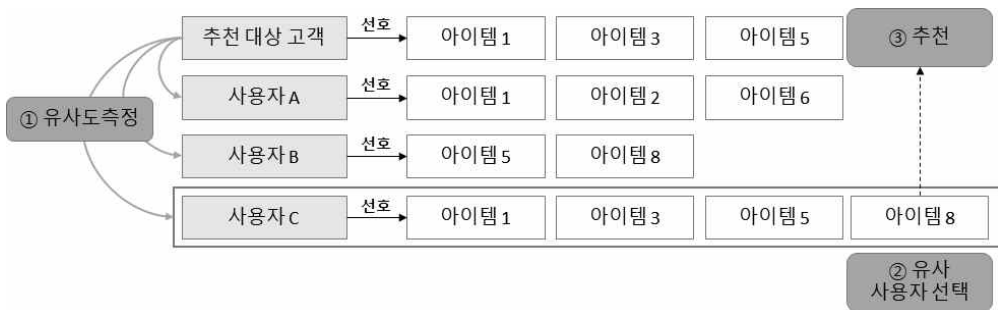


<그림 II-1> 추천 시스템의 분류

추천 시스템은 추천하는 기법에 따라 협업 필터링(Collaborative filtering), 내용 기반 필터링(Content-based filtering), 두 가지를 결합한 하이브리드(Hybrid) 형태의 3가지 접근법으로 분류할 수 있다.

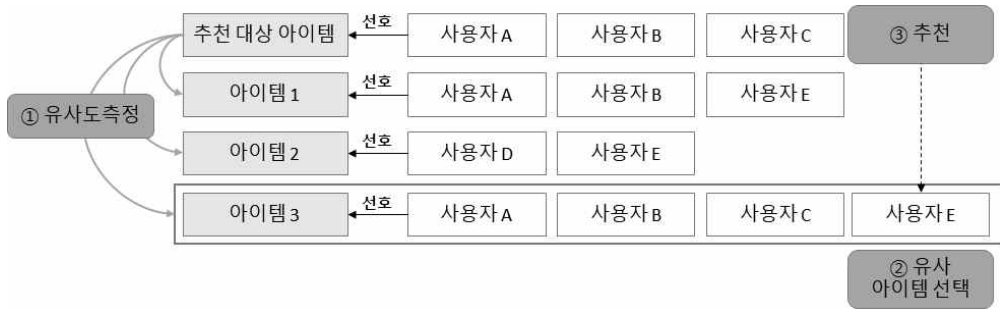
추천 시스템 중에서 가장 우수한 성능을 나타내고 있는 협업 필터링 추천 기법은 Goldberg et al. (1992) 연구에서 개념이 처음으로 정의되었다. 이후 미네소타 대학교 컴퓨터 공학과와 연구실의 GroupLens를 시작으로 인터넷 서점인 아마존(Amazon.com), 음악 관련 제품을 판매하는 온라인 쇼핑 웹 사이트를 운영하는 CDNow(CDNow.com), 영화 추천 사이트인 MovieFinder 등과 같은 다양한 분야에서 사용되고 있다(Resnick et al., 1994; Konstan et al., 1997; Schafer et al., 2001). 협업 필터링의 기본 아이디어는 특정 아이템에 대해 선호도가 유사한 고객들은 다른 아이템에 대해서도 비슷한 선호도를 보일 것을 바탕으로 아이템 간 또는 사용자 간 유사도를 기반으로 선호도를 예측하는 방법이다. 내용 기반 접근방식이 사용자와 아이템 정보에만 의존하여 선호도를 예측하는 반면

협업 필터링은 사용자가 아이템에 대해 평가한 정보를 사용해 선호도를 예측한다는 것이 가장 큰 차이점이다. 협업 필터링은 추천 대상 고객과 취향이 비슷한 사용자를 유사도를 측정하여 선정하고, 비슷한 사용자들이 선호하는 아이템을 추천 대상 고객에게 추천하기 때문에 추천되는 아이템의 다양성을 보장할 수 있다.



<그림 II-2> 사용자 기반의 협업 필터링 기법

<그림 II-2>는 사용자 기반 협업 필터링의 기본적인 개념을 나타내고 있다. 추천 대상 고객이 선택되면, 추천 대상 고객과 다른 사용자들 간의 구매 이력을 바탕으로 유사도를 측정한다. 추천 대상 고객과 구매한 아이템이 일치할수록 높은 유사도를 갖게 되며 취향이 비슷한 것으로 인식한다. 추천 대상 고객과 모든 사용자의 유사도를 측정했을 때 유사도가 가장 높은 사용자를 가장 취향이 비슷한 이웃으로 선택한다. 예를 들어, <그림 II-2>와 같이 추천 대상 고객과 취향이 가장 비슷한 사용자는 '아이템 1, '아이템 3, '아이템 5'를 구매한 '사용자 C'이다. 협업 기반 필터링의 마지막 단계에서는, 아이템 추천 단계로써 유사도 측정을 통해 선택된 '사용자 C'는 구매하였으나 추천 대상 고객은 아직 구매하지 않은 '아이템 8'을 최종적으로 선택하여 추천 대상 고객에게 추천한다.



<그림 II-3> 아이템 기반 협업 필터링 기법

<그림 II-3>은 아이템 기반 협업 필터링의 기본 개념으로써 추천 대상 아이템을 기준으로 유사한 아이템을 선정한 뒤, 선정된 ‘아이템 3’은 구매하였으나 추천 대상 아이템은 구매하지 않은 ‘사용자 E’에게 추천 대상 아이템을 최종적으로 추천하게 된다.

협업기반 필터링에서는 추천 대상 고객의 선호도를 예측하기 위해 사용자-아이템 행렬이 사용되며 행렬값은 사용자의 아이템에 대한 만족도를 점수로 나타낸다. <표 II-1>은 사용자의 아이템에 대한 만족도를 점수로 나타낸 사용자-아이템 행렬 예시이다.

<표 II-1> 사용자-아이템 행렬 예시

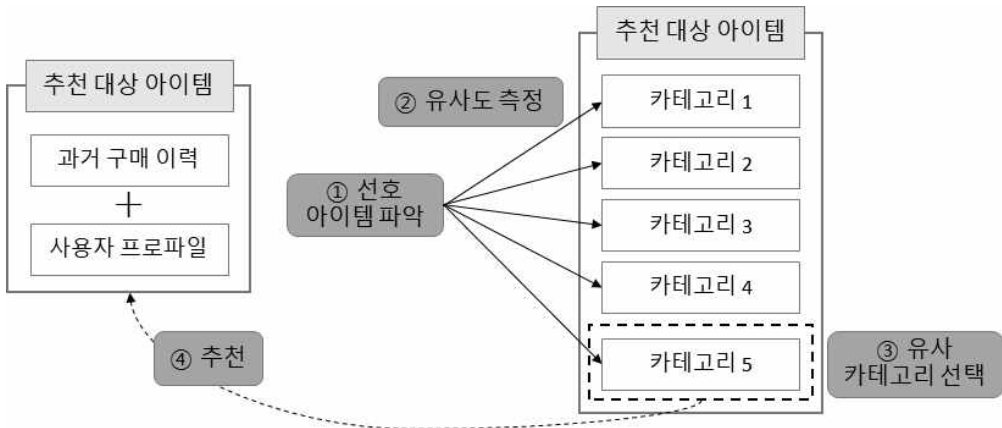
	아이템 1	아이템 2	아이템 3	아이템 4
사용자 A	4	—	5	—
사용자 B	4	2	1	—
사용자 C	3		2	—
사용자 D	4	4	—	—
사용자 E (신규 사용자)	—	—	—	—

<표 II-1>의 사용자-아이템 행렬에서의 각 상품의 인기에 따라 아이템 1과 같이 상품에 대해 많은 평가를 받을 수 있으며, 아이템 4와 같이 아직 아무런 평가를 받지 못하는 데이터 편중이 발생할 수 있다. 또한, 사용자 e와 같이 상품에 대해 평점값을 입력하지 않은 새로운 사용자가 존재할 수 있다. 이런 경우, 유사도를 측정할 만한 충분한 데이터가 존재하지 않기 때문에 선호도를 예측하는 것이 불가능하며 이를 콜드 스타트(Cold start) 라고 한다.

일자리 추천의 경우 사용자가 특정 채용정보에 대한 선호도에 따라 다른 사용자에게 채용정보를 추천할 수 있도록 사용자-아이템 행렬을 생성할 때 채용정보가 아이템에 해당하는데 채용정보에 대한 사용자의 만족도 점수를 반영하기 어렵다. 또한, 협업기반 필터링은 콜드 스타트 문제가 매우 민감하다. 특히 채용정보는 지원 기간이 정해져 있어 인기 있는 아이템을 추천해주는 방식인 협업기반 필터링 추천은 의미가 없고, 추천하더라도 새로운 채용정보에 대한 사용자의 만족도 또는 평가 점수가 없으므로 추천의 퀄리티가 낮아져 일자리 추천 방식으로 부적합하다. 일자리 추천은 다른 사용자의 아이템에 대한 선호도나 인기를 반영하지 않고 개인화된 사용자의 이력서 정보와 기업에서 요구하는 채용정보의 유사도를 측정해서 추천하는 내용 기반 추천이 올바르다.

내용 기반 필터링 방식은 정보검색기술에 바탕을 둔 시스템으로, 협업 필터링이 사용자의 행동 기록을 이용하는 반면, 콘텐츠 기반 필터링은 항목 자체를 분석하여 추천을 구현한다. 아이템의 내용을 직접 분석하여 아이템과 아이템 혹은 아이템과 사용자 선호도 간 유사성을 분석하여 이를 토대로 고객에게 아이템을 추천해주는 방식이다(Wu et al., 2000).

<그림 II-4>는 내용 기반 필터링 추천 기법을 보여주고 있다.



<그림 II-4> 내용 기반 필터링 추천 기법

사용자의 기본 정보나, 사용자가 아이템에 대해 평가한 점수, 별점 또는 과거 구매 이력을 바탕으로 생성된 정보인 사용자 프로필(User profile)을 통해 선호하는 아이템을 파악하고 미리 분류된 아이템 프로파일(Item profile)과 사용자의 선호 아이템 간의 유사도를 측정한다. 최종적으로, 유사도가 가장 높게 나타난 아이템을 추천 대상 사용자에게 추천한다(Lang, 1995). 즉, 내용 기반 필터링을 위해서는 항목을 분석한 아이템 프로파일과 사용자의 선호도를 추출한 사용자 프로파일을 추출하여 이의 유사성을 계산한다. 내용 기반 필터링 추천 기법은 분석이 쉽고 메타데이터를 이용할 수 있어 도서, 음악, 영화뿐만 아니라 텍스트 기반 뉴스 및 인터넷 기사 등을 추천하는데 널리 쓰이고 있다(Pazzani et al., 2007).

추천 시스템의 추천 기법에 대해서는 <표 II-2>를 통해 설명하였고, <표 II-3>은 추천 시스템의 추천 기법별 장점, 단점을 정리하였다.

<표 II-2> 추천 시스템의 추천 기법

추천 시스템	추천 기법
협업 필터링 (Collaborative Filtering)	대규모의 기존 사용자 행동 정보를 분석하여 해당 사용자와 비슷한 성향의 사용자들이 기존에 좋아했던 항목을 추천하는 기술
내용 기반 필터링 (Content-based Filtering)	사용자가 특정한 아이템을 매우 선호하는 경우, 그 아이템과 비슷한 콘텐츠를 가진 다른 아이템을 추천하는 기술
하이브리드 (Hybrid)	신규 콘텐츠들은 콘텐츠 기반 필터링으로 분석을 진행한 후 충분한 데이터가 쌓이게 된다면 협업 필터링으로 정확성을 높이는 방식으로 진행하는 협업 필터링과 내용 기반 필터링을 조합한 추천 기술

<표 II-3> 추천 시스템의 장·단점

추천 시스템	장점	단점
협업 필터링 (Collaborative Filtering)	<ul style="list-style-type: none"> 아이템 종류에 상관없이 추천 가능 특징을 선택할 필요 없음 (No feature selection, only rating vector) 직관적인 결과 항목의 구체적인 내용 분석 필요 없음 	<ul style="list-style-type: none"> 콜드 스타트(Cold Start): 기존에 없던 새로운 항목이 추가되는 경우는 추천 불가 희소성(Sparsity) 문제: 등급 행렬 (ratings matrix) 대부분 비어있어 동일한 아이템을 등급을 매긴 고객 찾기가 힘들 초기 평가자(First rater) 문제: 이전에 등급(ratings) 정보 없는 아이템 추천 불가 (새로운 아이템 또는 독특한 아이템) 인기 편향성(Popularity bias) 문제: 인기 있는 아이템만 추천하는 경향으로 독특한 취향 추천 불가

내용기반 필터링 (Content- based Filtering)	<ul style="list-style-type: none"> ◦ 다른 사용자에게 대한 데이터 필요 없음(No cold-start or sparsity problems) ◦ 개인의 독특한 취향을 고려한 추천이 가능 ◦ 새로운 아이템, 대중적이지 않은 아이템도 추천 가능(No first-rater problem) ◦ 사용자에게 추천하는 이유 설명 가능 (추천된 아이템의 내용적인 특징 정리) 	<ul style="list-style-type: none"> ◦ 유사도에 필요한 특징(feature) 선택 어려움(이미지, 영화, 음악 등) ◦ 새로운 사용자에게 추천 힘들(사용자 프로필생성이 어려움) ◦ 특화된 것들만 추천 가능(Overspecialization)
하이브리드 (Hybrid)	<ul style="list-style-type: none"> ◦ 콜드 스타트와 희소성 문제와 같은 추천시스템의 공통적인 문제 극복 가능 	<ul style="list-style-type: none"> ◦ 데이터 특성에 맞춰 다양한 방식으로 결합된 하이브리드 기법 설계하는 과정이 복잡함

일자리 추천에서는 개인의 독특한 취향을 고려한 추천과 새로운 아이템이나 대중적이지 않은 아이템도 추천이 가능한 내용 기반 추천을 많이 사용한다. 일자리 추천에서 내용 기반 필터링 추천은 사용자 자신만의 이력서의 정보와 기업의 채용정보의 특징을 기반으로 추천하기 때문에 새로운 채용정보에 대한 콜드 스타트 문제에 덜 예민하다. 그러나 내용 기반 필터링 추천은 <표 II-3>에서 내용 기반 추천의 단점을 설명한 것과 같이 추천 대상 사용자의 선호도를 파악하기 위해서 자신만의 과거 활동 이력이나 프로파일 정보가 이용되는데 과거 활동 이력이 부족한 경우 추천 성능을 보장할 수 없으며, 활동 이력이나 사용자 프로파일 정보가 모두 존재하지 않는다면 추천 시스템 구현이 불가능 하다(Bobadilla et al., 2013). 또한, 신규 사용자가 가입했을 경우, 필요한 사용자의 이력서 정보가 없거나 내용이 부실하게 되면 콜드 스타트 문제가 발생할 수 있다.

콜드 스타트 문제를 해결하기 위해, 초기 연구들은 대부분 사용자 및 품목의 메타정보로서 범주형 정보를 이용하였다. 범주형 정보는 사용자 및 품목의 메타정보 중 사용자의 성별, 직업, 국적, 품목의 장르 및 카테고리, 문서 내의 단어 등을 의미하며 이 정보들은 사용자와 품목을 위한 이원 벡터 모델 또는 단어 사전을 이용한 벡터 모델로 쉽게 표현할 수 있다. 그러나 이 방법 또한 내용 기반 추천 시 단어 사이의 문맥이나 숨은 의미를 반영할 수 없는 한계점이 있다.

2) 일자리 추천 시스템

기존 일자리 추천 연구는 구직자의 프로파일과 채용정보의 요구사항을 추천에 활용하는 내용 기반 필터링 방식으로 추천이 이뤄졌다.

Valverde-Rebaza et al.(2018) 연구에서는 구직자의 이력서 정보를 모두 사용하지 않고 일자리 추천에 영향을 미치는 구직자의 전문 기술 정보만을 추출하여 워드 임베딩을 얻는 방식을 사용하면 추천 성능에 영향을 줄 수 있다고 제안하였다. 전문기술에 대한 용어추출 및 표현방식에 따라 벡터공간모델 기반의 워드 임베딩 방식(Word2Vec)과 빈도기반의 TF-IDF(Term Frequency - Inverse Document Frequency) 방법을 사용하여 각각 성능을 비교하였다. IT 분야의 기술에만 국한하여 일자리 매칭 커뮤니티 링크드인(LinkedIn)에서 50명의 프로파일을 수집하고 Catho에서 3877개의 일자리 프로파일을 수집하여 벡터 공간모델 기반의 워드 임베딩 방식의 두 가지 방식인 CBOW(Continuous Bag Of Word)와 Skip-gram을 실험한 결과, Skip-gram 임베딩을 방식을 활용한 추천이 가장 좋은 성능을 보였다고 보고하고 있다.

Le & Mikolov(2014)는 TF-IDF 및 bag-of-words와 같은 워드 임베딩 방식 모형은 자연어 처리작업에 효과적이지만 단어의 출현 빈도에만 초점을 두기 때문에 맥락을 이해하는 데는 부족하다고 지적하며, 맥락을 이해하도록 하기 위해서는 GloVe(Global Vectors for Word Representation), ELMo(Embeddings from Language Model), Word2vec 등과 같은 임베딩 모형을 활용하여 벡터화할 것을 권고하고 있다.

Leksin & Ostapets(2016) 연구에서는 일자리 추천에 있어서 구직자와 일자리에 대한 상호작용이 없는 데이터가 많아 협업 필터링으로만 해결하기 어렵다 하여, 행렬분해 기법(Factorization Machine)과 아이템 기반 협업 필터링(item-based collaborative filtering), 내용 기반 토픽 모델을 결합한 하이브리드 추천 기법을 제안하였다. 구직자와 일자리의 프로파일로부터 추출된 키워드(토큰)가 성능향상을 위한 중요한 특성으로 작용함을 보여주었다. 또한, Pessemier et al.(2016) 연구에서는 구직자의 이력서와 관심 채용정보 클릭 여부 등의 지원 이력 및 관심도를 토대로 사용자가 선호하는 사용자 프로파일을 생성하였다. 구직자가 클릭하거나 지원한 채용정보에서 제목, 경력 수준, 분야, 산업, 국가, 지역, 위치, 고용 및 태그를 속성으로 추출하여 프로파일에 적용하였다. 사용자의 프로파일-미지의 채용정보와의 유사도 점수를 기반으로 추천을 수행하는 내용 기반 추천(Content-based)과 사용자 지원 이력 및 관심도의 빈도로 가중 평균하는 KNN(K-Nearest Neighbor)기반 추천을 결합하여 하이브리드 일자리 추천 방법을 제안하였다. 채용정보에서 사용한 속성 정보, 채용정보 클릭 여부 및 지원 이력의 데이터인 상호작용 정보가 없는 콜드 스타트 사용자의 경우(실험 데이터 구직자의 26.5% 상호작용 정보 없음) 내

용 기반 추천 알고리즘을 적용하고 이력서가 없는 신규 사용자의 경우 하이브리드 추천 알고리즘에서 KNN 알고리즘으로 대체 하였다. 내용 기반 및 KNN 접근방식을 결합한 하이브리드 추천은 구직자의 이력서 정보 및 상호작용 데이터가 빠진 사용자 추천에 대한 콜드 스타트 문제를 처리할 수 있었다.

Maheshwary & Misra(2018) 연구에서는 구직자의 반구조화된 (Semi-structured) 이력서를 채용정보의 직무설명과 매칭하여 구직자에게 적절한 일자리를 추천하는 삼 신경망 기반의 일자리 추천모형을 제안 하였다. 이처럼 기존의 일자리 추천 연구에서는 구직자의 이력서 정보와 채용정보를 기계학습 모형에 반영하는 전처리 과정에서 다양한 벡터화 기법에 초점을 맞춰 연구가 진행되고 있다. 그러나 궁극적으로 일자리 추천은 구직자와 기업 모두에게 필요한 일자리와 인재를 추천해주는 양방향 추천이 필요하다.

송희석(2020)의 연구에서는 상호작용 이력이 있는 취업공고와 구직자 이력서 사이의 적합도를 학습하여 특정 구직자의 이력서에 적합한 일자리 또는 특정 취업공고에 적합한 구직자를 매칭하는 양방향 척도학습모형을 제안하였다.

이 연구에서는 구직자ID와 취업공고ID 대신 구직자 프로파일과 취업공고문서의 텍스트를 bag-of-words 벡터로 변환하여 동시에 입력한 후심층 신경망을 거쳐 최종 출력층에서 적합도 여부를 학습하는 내용 기반 척도학습모형을 제안하였는데 제안된 방법은 구직자 프로파일과 취업공고문서를 모형에 입력하기 위해 bag-of-words 벡터로 변환하는 전처리

과정을 거친다. 전처리 과정을 거친 구직자 프로파일과 취업공고 문서로 입력된 벡터는 각각 별도의 임베딩 층을 거친 후 내적(Dot Product)을 통해 결합하여 심층신경망을 통과하게 된다. bag-of-words 입력층 위에 임베딩층을 둔 것은 같은 구직자들이 지원한 일자리들을 유사한 일자리 벡터로 표현하거나 같은 일자리에 지원한 구직자 프로파일을 유사한 구직자 프로파일 벡터로 투사함으로써 학습결과 입력 벡터에 시맨틱을 부여하기 위함이다. 최종 출력층은 해당 구직자 프로파일 벡터와 일자리 벡터 간 지원 이력 등의 지원 이력 내역을 나타내는 한 개의 노드로 구성되며 지원 이력 내역이 있으면 1 없으면 0을 출력하도록 학습이 이루어진다.

내용 기반 척도학습 모형은 협업 필터링 기반의 추천모형과 달리 신규 구직자 또는 신규 일자리공고에 대해서도 추천이 가능한 장점이 있다.

<표 II-4>는 일자리 추천 시스템의 기존 연구내용을 비교 분석한 자료이다.

<표 II-4> 일자리 추천 시스템 비교

연구내용	추천 기법	추천 대상 (Target)	연구자
일자리 추천에 영향을 미치는 구직자의 전문기술 정보만을 추출하여 워드 임베딩(Skip-gram)을 활용한 추천 방식 제안	<ul style="list-style-type: none"> ◦ 내용기반 ◦ 워드 임베딩(Skip-gram) 	<ul style="list-style-type: none"> ◦ 일자리 추천 ◦ 단방향 	Valverde-Rebaza et al.(2018)
행렬분해 기법과 아이템 기반 협업 필터링, 내용 기반 토픽 모델을 결합한 하이브리드 일자리 추천 제안	<ul style="list-style-type: none"> ◦ 하이브리드 	<ul style="list-style-type: none"> ◦ 일자리 추천 ◦ 단방향 	Leksin & Ostapets (2016)

사용자의 프로파일-채용정보와의 유사도 점수를 활용한 내용기반 추천과 사용자 지원 이력 및 관심도의 빈도로 가중 평균하는 KNN 추천을 결합하여 하이브리드 일자리 추천 제안	◦ 하이브리드	◦ 일자리 추천 ◦ 단방향	Pessemier et al.(2016)
구직자의 이력서를 채용공고의 직무설명과 매칭하여 구직자에게 적절한 일자리를 추천하는 삼 신경망 기반의 일자리 추천 제안	◦ 삼 신경망 ◦ 워드 임베딩 (Doc2Vec)	◦ 일자리 추천 ◦ 단방향	Maheshwary & Misra(2018)
양방향의 인재매칭을 위해 구직자 정보와 일자리 정보가 동시에 입력되고 최종 출력층에서 적합도가 출력되도록 하는 척도학습(metric learning) 모델 제안	◦ 내용 기반 ◦ 척도학습 ◦ 워드 임베딩 (BoW)	◦ 일자리 추천 ◦ 구직자 추천 ◦ 양방향	송희석(2020)

<표 II-4>의 일자리 추천 기존 연구 비교 분석한 것과 같이 하이브리드 기법과 삼신경망 추천 기법은 모두 양방향 추천이 불가능하였다. 양방향 추천이 가능한 척도모형 또한, 구직자 프로파일과 잡공고에서 수집한 데이터만을 활용하여 직접 훈련을 통해 임베딩 벡터를 획득하고 있어서, 다양한 전문 용어에 대한 의미적 유사성을 모두 반영할 수 없는 단점이 있다.

이러한 단점을 해결하기 위해 풍부한 단어들로 사전 훈련된 언어 모델을 활용한 전이학습(Transfer Learning)이 주목받고 있다. 최근 머신러닝 및 딥러닝 기술이 가장 많이 적용되고 있는 분야 중 하나가 자연어 처리(NLP, Natural Language Processing) 분야이다. 기존 자연어 처리 분야 연구에서는 워드 임베딩 모델을 활용하여 모델의 입력하기 위한 벡터

화(Vectorization)를 주요 연구로 진행하였으나(Le & Mikolov , 2014), 컴퓨터 비전(Computer Vision) 및 이미지 처리(Image processing) 분야에서 활용하여 좋은 성과를 내는 전이학습 모델을 자연어 처리 분야에서도 전이 학습된 언어 모델을 이용하는 것이 성능향상에 도움이 된다고 보고하고 있다(유소엽 & 정옥란, 2019).

여러 연구자가 다양한 각도에서 일자리매칭을 위해 연구를 진행하고 있으나, 실제 일자리매칭을 위한 다양한 커뮤니티 포털에서 사용하는 일자리매칭 기술은 아직도 대부분이 단순 키워드 검색에 머물러있다. 최근 일자리매칭을 위해 발전된 기술을 활용하고자 하는 시도들이 나타나고 있다. LinkedIn(linkedin.com)이 개인화된 일자리 추천 서비스 도입에 가장 적극적이다(Kenthapadi et al., 2017). 그러나, LinkedIn의 일자리 추천 시스템은 프로파일 내의 구조화된 정보에 기반을 둔 검색기반의 매칭시스템으로 볼 수 있다.

국내에서는 민간 기업 중심의 AI 기반 구직 매칭 솔루션이 잇달아 출시되고 있지만, 대부분은 구직자의 정보 분석을 기반으로 한 구직 매칭 서비스이다. 사람인(saramin.co.kr) 구인 구직 포털 사이트의 AI 기반 채용 추천 서비스(AVATAR SEARCH 솔루션)는 구직자의 직종, 지역, 경력 등 구직자의 정보를 분석하고 개인에 최적화된 채용공고를 추천한다. 또한, 정부는 지역 단체 및 민간 취업 사이트와 연계된 고용 정보망(Worknet, Employment Information Network work.go.kr)을 통해 맞춤형 취업 정보, 취업 상담, 취업 지원 서비스를 제공하고 있다. ‘인공지능 기반 일자리매칭’ 서비스를 통해 이력서, 표준직무기술서 등을 활용한 내용 기반 필터링 기법과 직무역량, 구인기업 속성의 가중치를 결합한 하

이브리드 추천 기법을 제공한다. 최근에는 직업·인재 정보를 신설하여 일자리, 자격, 회사, 직업 심리 검사 등과 관련 콘텐츠를 구직 관련 빅데이터 분석을 통해 제공했다. 그러나 직업·인재 정보는 여전히 단순하고 제한적이며 개인 정보 이용의 어려움으로 실제 매칭에 한계가 있다(한역수 & 심진보, 2019).

SK C&C(recruit.skcc.co.kr)의 AI 기반 문서 분석 솔루션은 회사의 채용공고에 지원한 사람들의 자기소개 키워드와 지원자의 성향, 채용과 관련된 양적, 질적 평가를 분석한다(한역수 & 심진보, 2019).

민간 기업 중심의 AI 기반 구직 매칭 솔루션은 대기업 위주의 원하는 인재 확보와 대규모 중심의 효율적인 채용 프로세스에 중점을 두고 있다. <표 II-5>는 AI 기반 구직 매칭 솔루션을 정리하였다.

<표 II-5> AI 기반 구직 매칭 솔루션

업체명	AI 기반 솔루션 주요 내용	홈페이지
LinkedIn	◦ 일자리 추천 시스템은 프로파일 내의 구조화된 정보에 기반한 검색기반의 매칭 시스템	www.linkedin.com
사람인 (Saramin)	◦ AI 기반 채용 추천 서비스 (AVATAR SEARCH 솔루션) ◦ 입사 지원한 수천만 건의 이력서를 분석하여 이력서별 아바타를 매칭하여 최적의 채용공고를 추천	www.saramin.co.kr
워크넷 (Worknet)	◦ 인공지능 기반 일자리매칭 서비스 ◦ 빅데이터로 축적된 개인의 경력, 교육·훈련, 자격정보 등을 통해 구직자에게 최적의 일자리를 추천	www.work.go.kr

SK C&C	<ul style="list-style-type: none"> ◦ AI 기반 문서 분석 솔루션(Aibril HR) ◦ 채용서류 평가 서비스 ◦ 자기소개 키워드와 채용 관련 지원자의 성향, 정량, 정성 평가를 분석 	recruit.skcc.co.kr
--------	---	--------------------

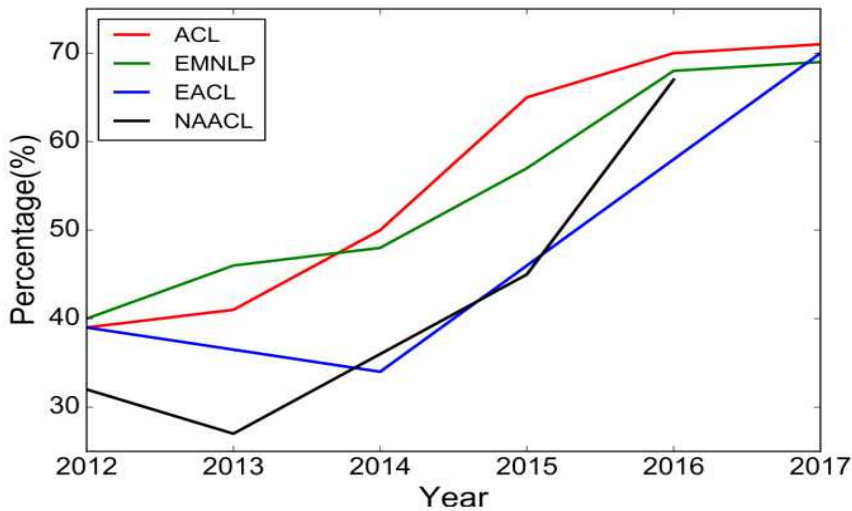
기업에서는 한 기업에서만 사용 할 수 있는 자체적인 인재 채용시스템을 구축하고, 구직자는 여러 개의 기업 채용시스템 및 취업 포털 시스템에 맞춰 구직을 준비하고 있는 상황으로 일자리 검색 및 정보 미스매치 현상은 더욱 심화하고 있다. 또한, 구인 구직 포털 사이트에서의 딥러닝 기술을 적용과 관련하여 정확하게 어떤 기술을 적용하고 있는지 공개하지 않고 있으며, 일자리 추천 기법들이 활발하게 연구되고는 있지만, 연구 결과로 나온 추천 기법을 적용하여 실제로 구직자들이 사용할 수 있는 채용 시스템이나 취업 포털 시스템으로 구축되는 경우는 드물다.

2. 자연어 처리

1) 자연어 처리

자연어 처리(NLP; Natural Language Processing) 분야는 인간 언어 분석과 표현(Representation)을 자동화하기 위한 계산 기법이다. 자연어 처리 연구는 문장 하나 처리하는 데 7분이 소요되던 천공카드(Punch Card)와 배치 과정(batch processing)의 시대로부터 1초 만에 수백만 웹페이지를 처리할 수 있는 구글의 시대로 발전해왔다(Cambria & White, 2014). 자연어 처리 분야는 컴퓨터가 내가 원하는 데이터를 특정 패턴이나 순서로 추출해 가공하는 구문분석(Parsing), 품사 태깅(POS Tagging, Part-of-Speech Tagging)에서부터 기계 번역, 대화 시스템에 이르기까지 모든 과업을 수행할 수 있도록 한다.

딥러닝 아키텍처와 알고리즘은 컴퓨터 비전과 패턴인식 같은 분야에서 이미 의미 있는 발전을 이뤘다. 이러한 경향이 나타난 이후, <그림 II-5>과 같이 최근 딥러닝 기법 기반의 자연어 처리 연구가 늘고 있다(Young, et al., 2018).

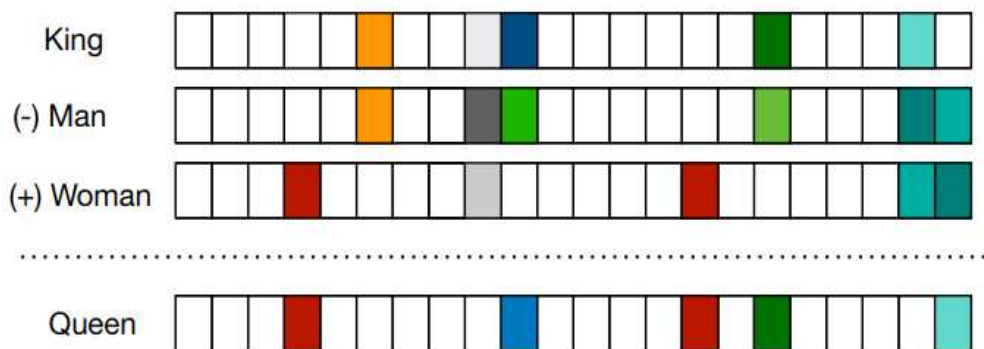


<그림 II-5> 최근 6년 간 ACL, EMNLP, EACL, NAACL에 게재된 딥러닝 논문의 비율(%), 출처: Young, et al.(2018)

지난 수십 년간 자연어 처리 문제를 풀기 위한 머신러닝의 접근은 고차원이면서 희소 한(Sparse)의 특징(Feature)을 학습한 ‘얕은 모델(Shallow models, 예: SVM/로지스틱 회귀)’에 기반을 둔 것이다. 최근 수년간 단어 밀집 벡터(Dense vector representation)에 기반을 둔 인공신경망(Neural Network)이 다양한 자연어 처리 작업(NLP task)에서 우수한 성능을 보여줬다. 이러한 추세는 워드 임베딩(Milokov et al., 2010, 2013)과 딥러닝 기법(Socher et al., 2013)의 성공에 힘입은 것이다. 딥러닝은 자동화된 특징 추출 및 표현(Multi-level automatic feature representation learning)을 가능하게 한다. 그러나 전통적인 머신러닝에 기반을 둔 자연어 처리 시스템은 사람이 직접 추출한(Hand-crafted) 특징에 강하게 의존한다. 이러한 특징들은 추출하는 데 시간이 많이 소요되고 많은 경우 불완전하다.

Collobert et al.(2011)은 간단한 딥러닝 프레임워크를 제시했다. 이 프레임워크는 개체명 인식(NER, Named Entity Recognition), 의미역 결정(SRL, Semantic Role Labeling), 품사 태깅(POS tagging) 같은 일부 NLP 태스크에서 최첨단 기법이다. 그 이후 수많은 복잡한 딥러닝 기반의 알고리즘이 어려운 자연어 처리 문제를 풀기 위해 제안되었다. 합성곱 신경망(CNN, Convolutional Neural Network), 순환신경망(RNN, Recurrent Neural Network), 재귀신경망(RNNR, Recursive Neural Network, Recursive) 등 주요 딥러닝 모델이 등이 대표적이다.

자연어 처리를 위해서는 임베딩(Embedding)이란 자연어 처리를 위해서는 텍스트를 기계가 문장에서 단어가 쓰인 의미를 문맥을 통해 구분하고, 이해할 수 있도록 단어를 0과 1의 수치로 표현하는 방법으로 벡터화(Vectorization) 라고도 한다(Sohrabi, et al., 2018). 임베딩은 전체 단어 간의 관계에 맞춰 해당 단어의 특성을 갖는 벡터로 바꿔주므로 단어들 사이의 유사도를 계산하는 기법이다. 이러한 유사도 계산을 통해 단어 간의 의미적·문법적 관계를 파악해낼 수 있다. <그림 II-6>과 같이 Man-Woman 사이의 관계와 King-Queen 사이의 의미 차이가 임베딩에 함축되어 있으면 좋은 임베딩이라 할 수 있다.



<그림 II-6> 임베딩(Embedding) 예시, 출처: Young, et al.(2018)

임베딩 기법의 발전 흐름과 종류는 통계적 기반과 신경 네트워크(Neural network) 기반으로 나눌 수 있고 단어 수준과 문장 수준의 임베딩 기법으로 구분할 수 있다. 임베딩 초기 기법은 통계적 기반을 중심으로 말뭉치라 불리는 코퍼스(Corpus)의 통계량을 직접 활용하였다.

<표 II-6>은 통계적 기반 임베딩 기법을 정리한 것이다.

<표 II-6> 통계적 기반 임베딩 기법

임베딩 기법	설명
TDM (Term-Document Matrix)	<ul style="list-style-type: none"> ◦ 단어-문서행렬이라고 부르며 문서에서 등장하는 단어들의 빈도를 행렬로 표현하는 것으로 ◦ 단어가방(BoW bag-of-word)의 표현을 행렬로 표현 ◦ 장점: 수치화된 단어들을 서로 비교 가능 ◦ 단점: 고려해야 할 단어 수가 대량일수록 적용하는데 한계
TF-IDF (Term Frequency-Inverse Document Frequency)	<ul style="list-style-type: none"> ◦ 특정 단어가 문서 내에서 출현하는 빈도(TF)값과 흔한 단어는 문서에서 자주 등장되는 경우가 많아 역빈도(IDF)값을 계산하는 것 ◦ TF-IDF값이 높은 단어 일수록 문서에서 중요도가 높다고 간주하여, 문서의 핵심어 추출, 검색 결과의 우선순위 결정 등에 이용
One-hot Encoding	<ul style="list-style-type: none"> ◦ 문서의 단어를 벡터로 표현하는 방식으로 '0' 과 '1' 로 구분하는 방법 ◦ 장점: 텍스트를 유의미한 숫자(벡터)로 바꾸는 가장 손쉬운 방법론 ◦ 단점 : <ul style="list-style-type: none"> - 단어 갯수가 늘어날수록 벡터 저장 공간이 늘어나야 함 - 단어의 문맥정보가 사라짐 - 단어 간 유사도를 파악할 수 없음 - 유사성에서 반대되는 의미를 반영하지 못함

워드 임베딩의 역사는 인공 망을 이용하여 주변 단어의 단어 등장 확률을 예측한 Neural Probabilistic Language Model(NPLM)이 발표된 이후부터 Word2Vec → FastText → ELMO → BERT 기법으로 발전하고 있다. 가장 최신의 언어분석 기법인 BERT는 다른 언어분석 기법들과 비교하면 임베딩 결과에서 우수한 성능을 보인다. 이는 기존의 임베딩은 문장에서 단어를 순차적으로 입력받고 다음 단어를 예측하는 한 방향(uni-directional) 이지만 BERT는 문장 전체를 입력받고 단어를 예측하고 양방향(bi-directional) 학습이 가능하기 때문이다. 이러한 신경 네트워크 구조의 유연성과 풍부한 표현력으로 자연어의 문맥을 상당 부분 학습할 수 있고 높은 정확도를 보인다(유승의, 2021).

<표 II-7>은 신경 네트워크 기반 임베딩 기법을 정리한 것이다.

<표 II-7> 신경 네트워크 기반 임베딩 기법

임베딩 기법	설명
Word2Vec (Mikolov, et al., 2013)	의미적인 성질이 유사한 단어들은 벡터 공간상에서 유클리디안(Eucledian) 거리 나 코사인 유사도(cosine similarity) 거리가 가까운 벡터들로 표현되는 단어 간 유사도를 반영하고 단어를 벡터화할 수 있는 방법
FastText (Kuyumcu, et al., 2019)	단어를 개별 단어가 아닌 n-gram의 characters (Bag-Of-Characters)를 적용하여 임베딩하므로 하나의 단어를 여러 개로 잘라서 벡터로 계산하는 방식
ELMo (Peters, et al., 2018)	ELMO(Embeddings from Language Model)언어 모델로 하는 임베딩이란 새로운 워드 임베딩 방법론으로 사전 훈련된 언어 모델(Pre-trained Language Model)을 사용특징

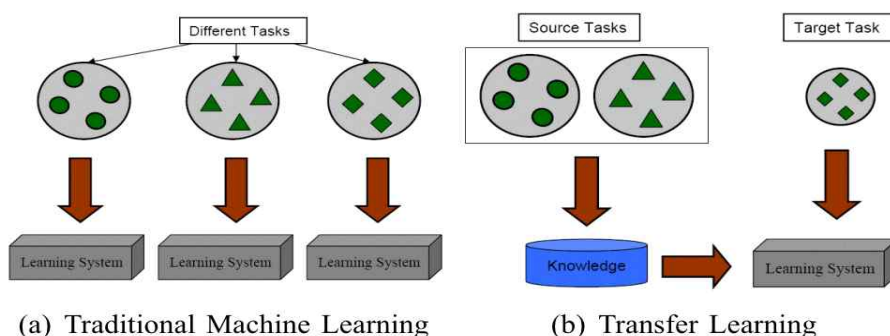
2) 전이 학습(Transfer Learning)

기존의 데이터 마이닝 및 기계학습 알고리즘은 이전에 수집된 레이블이 지정되거나 레이블이 지정되지 않은 훈련 데이터에 대해 훈련된 통계 모델을 사용하여 미래 데이터에 대한 예측을 수행한다(X. Yin, et al., 2006; Kuncheva, L.I., & Rodriguez, J.J., 2007; E. Baralis, et al. 2008).

준 지도(Semi-supervised) 분류(Zhu, 2006; Nigam, et al., 2000; Blum & Mitchell, 1998; Joachims, 1999)는 많은 양의 레이블이 지정되지 않은 데이터와 소량의 레이블이 지정된 데이터를 사용하여 레이블이 지정된 데이터가 너무 적어 좋은 분류기를 구축할 수 없다는 문제를 해결한다. 불완전한 데이터셋(Imperfect datasets)에 대한 지도 및 준 지도 학습의 연구가 진행되었다. Zhu & Wu(2006) 연구에서는 잡음이 많은 클래스 레이블(Noisy classlabel) 문제를 처리하는 방법을 제시했다. 향후 샘플에 대해 추가 테스트를 수행할 수 있는 경우 비용에 민감한 학습으로 간주하였다(Yang et al., 2006). 그런데도 대부분은 레이블이 지정된 데이터와 레이블이 지정되지 않은 데이터의 분포가 동일하다고 가정한다.

이와 대조적으로 전이학습은 훈련 및 테스트에 사용되는 도메인, 작업 및 배포를 다르게 할 수 있다. 현실 세계에서 우리는 전이학습의 많은 예를 관찰할 수 있다. 사람들이 이전에 배운 지식을 지능적으로 적용하여 새로운 문제를 더 빨리 또는 더 나은 솔루션으로 해결할 수 있다는 사실이 전이학습 연구에 대한 동기 부여가 되었다. 기계학습 분야에서 전이학습의 근본적인 동기는 “배우기 위해 배우기(Learning to Learn)”에 대

한 NIPS-95(<http://socrates.acadiau.ca/courses/comp/dsilver/NIPS95LTL/transfer.workshop.1995.html>) 워크숍에서 논의되었다. 이전에 학습된 지식을 유지하고 재사용하는 지속적 기계학습(Lifelong Machine Learning; LML)방법의 필요성에 초점을 맞췄다. <그림 II-7>은 전통적인 기계학습 과정과 전이학습 과정의 차이점을 보여준다.



<그림 II-7> 기계학습 과정과 전이학습 과정 비교,

출처: Sinno Jialin Pan et al.(2009)

전통적인 기계학습 기술은 처음부터 각 작업을 배우려고 시도하는 반면, 전이학습 기술은 일부 이전 작업의 지식을 대상 작업에 전달(Transfer)하려고 시도한다. 전이학습 방법은 데이터 마이닝(예: ACM KDD, IEEE ICDM 및 PKDD), 기계학습(ICML, NIPS, ECML, AAAI, IJCAI, for example) 및 기계학습 및 데이터 마이닝의 응용 분야(ACM SIGIR, WWW and ACL for example)에서 여러 주요 분야에 사용되고 있다.

딥러닝은 새로운 분류 플랫폼으로 최근 연구자들의 주목을 받으며 많은 영역에 성공적으로 적용되고 있다. 생물 정보학(Bioinformatics) 및 로봇 공학(Robotics)과 같은 일부 영역에서는 데이터 셋에 레이블을 작성하는

데이터 어노테이션(Data Annotation) 작업 및 데이터 수집은 큰 비용이 발생한다. 이로 인해 어노테이션이 잘 달린 대규모 데이터 셋을 구성하는 것이 매우 어렵고, 이로 인해 개발이 제한된다(Chuanqi Tan, et al., 2018).

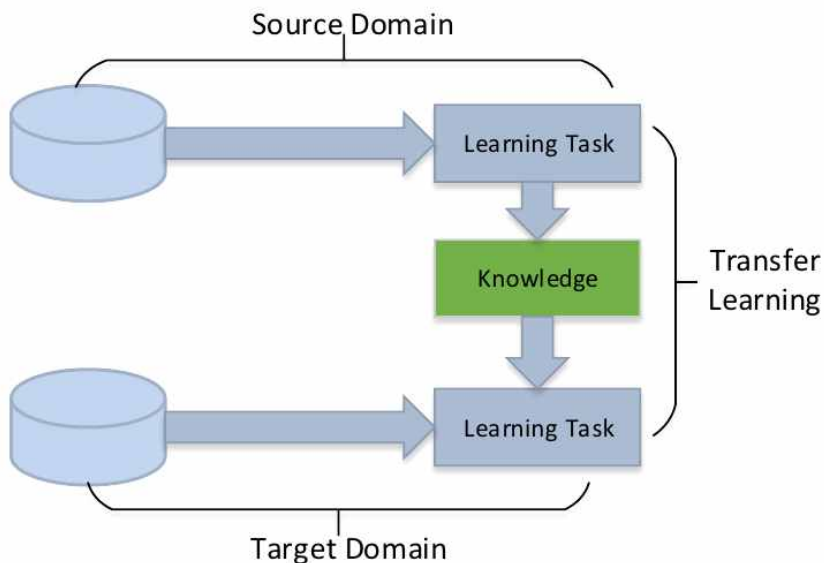
데이터 의존성(Data dependence)은 딥러닝에서 가장 심각한 문제 중 하나이다. 딥러닝은 데이터의 잠재된 패턴을 이해하기 위해 많은 양의 데이터가 필요하므로 기존 기계학습 방법에 비해 방대한 훈련 데이터에 대한 의존도가 매우 높다. 흥미로운 현상은 모델의 규모와 필요한 데이터의 크기가 거의 선형 관계에 있다는 것이다. 수용 가능한 설명은 특정 문제에 대해 모델의 표현 공간이 데이터 아래의 패턴을 발견할 수 있을 만큼 충분히 커야 한다는 것이다. 딥러닝 모델의 상위 계층(Pre-order layers)은 훈련 데이터의 특징(Features)을 식별할 수 있고 후속 계층(Subsequent layers)은 최종 결정을 내리는 데 도움이 되는 정보를 식별할 수 있다.

불충분한 훈련 데이터(Insufficient training data)는 일부 특수 영역에서 피할 수 없는 문제이다. 데이터 수집은 복잡하고 비용이 많이 들기 때문에 대규모의 고품질 어노테이션 데이터 셋을 구축하기가 매우 어렵다. 예를 들어, 생물 정보학 데이터셋의 각 샘플은 종종 임상 시험 정보 및 아픈 환자 정보들이다. 또한, 고가의 비용을 지불하고 훈련 데이터 셋을 구하더라도 구식이 되기 쉬우므로 새로운 작업에 효과적으로 적용할 수 없다.

전이학습은 훈련 데이터가 테스트 데이터와 독립적이고 동일하게 해야 한다는 독립 항등분포(Independent and identically distribution; i.i.d.) 가설을 완화하여 불충분한 훈련 데이터 문제에 대해 전이학습을 사용하도

록 동기를 부여한다. 전이학습에서 학습 데이터와 테스트 데이터는 독립
 항등분포일 필요가 없으며 대상 도메인의 모델은 처음부터 학습할 필요가
 없으므로 대상 도메인의 학습 데이터 수요와 학습 시간을 크게 줄일 수
 있다. 전이학습은 부족한 훈련 데이터라는 기본적인 문제를 해결하기 위
 한 기계학습의 중요한 도구이다. 훈련 데이터와 테스트 데이터가 독립 항
 등분포인 불충분한 훈련 데이터로 인해 개선하기 어려운 많은 영역에 큰
 긍정적인 영향을 줄 것이다.

딥러닝에 적용되는 전이학습의 학습 과정은 <그림 II-8>과 같다.



<그림 II-8> 전이학습 과정, 출처: Chuanqi Tan et al.(2018)

Pan, S.J., et al.(2009)과 Weiss, K., et al.(2016)의 조사에서는 전이
 학습 방법을 소스 도메인(Source domain)과 대상 도메인(Target
 domain) 간의 관계에 따라 세 가지 주요 범주로 구분하며 널리 받아들여
 지고 있다. 이러한 조사는 전이학습에 대한 과거 작업을 잘 요약한 것으

로, 여러 가지 고전적인 전이학습 방법을 도입했다. 또한, 최근에 더 새롭고 더 나은 방법이 많이 제안되었다.

최근 몇 년 동안 전이학습 연구 커뮤니티는 주로 도메인 적응 및 다중 소스 도메인 이전의 두 가지 측면에 중점을 두고 있었다. 오늘날 딥러닝은 최근 몇 년 동안 많은 연구 분야에서 지배적인 상황을 달성했다. 다음과 같이 정의된 심층 전이학습(Deep Transfer Learning)이라고 하는 심층 신경망(Deep neural networks)을 통해 지식을 효과적으로 전달하는 방법을 찾는 것이 중요하다. 심층 전이학습은 심층 신경망에서 다른 분야의 지식을 활용하는 방법을 연구한다. 심층 신경망이 다양한 분야에서 대중화되면서 이를 분류하고 요약하는 것이 매우 중요하다는 심층 전이학습 방법이 상당히 많이 제안되었다. 심층 전이학습에 사용되는 기술을 기반으로 심층 전이학습을 인스턴스 기반(Instances-based) 심층 전이학습, 매핑 기반(Mapping-based) 심층 전이학습, 네트워크 기반(Network-based) 심층 전이학습, 적대적 기반(Adversarial-based) 심층 전이학습의 네 가지 범주로 분류한다. 심층 전이학습 네 가지 범주로 분류한 내용을 <표 II-8>로 정리하였다.

<표 II-8> 심층 전이학습의 분류

심층 전이학습	설명	관련 연구
인스턴스 기반 (Instances-based)	적절한 가중치로 소스 도메인의 인스턴스를 활용	<ul style="list-style-type: none"> ◦ DAI, W., et al.(2007) ◦ Yao, Y., et al.(2010) ◦ Pardoe, D. & Stone, P (2010) ◦ Wan, C., et al.(2011) ◦ Li, N., et al.(2017) ◦ Xu, Y., et al.(2017) ◦ Liu, X., et al.(2017)

매핑 기반 (Mapping- based)	두 도메인의 인스턴스를 더 나은 유사성을 가진 새 데이터 공간으로 매핑	<ul style="list-style-type: none"> ◦ Tzeng, E., et al.(2014) ◦ Long, M., et al.(2015) ◦ Gretton, A., et al.(2012) ◦ Long, M., et al.(2017) ◦ Arjovsky, M., et al.(2017)
네트워크 기반 (Network- based)	네트워크 기반 소스 도메인에서 사전 훈련된 네트워크의 일부를 재사용	<ul style="list-style-type: none"> ◦ Huang, J.T., et al.(2013) ◦ Oquab, M., et al.(2014) ◦ Long, M., et al.(2016) ◦ Zhu, H., et al.(2016) ◦ Chang, H., et al.(2017) ◦ George, D., et al.(2017) ◦ Yosinski, J., et al.(2014)
적대 기반 (Adversarial -based)	적대 기반 적대적 기술을 사용하여 두 도메인에 모두 적합한 양도 가능한 기능을 찾음	<ul style="list-style-type: none"> ◦ Ajakan, H., et al.(2014) ◦ Ganin, Y., et al.(2015) ◦ Tzeng, E., et al.(2015) ◦ Tzeng, E., et al.(2017) ◦ Long, M., et al.(2017) ◦ Luo, Z., et al.(2017)

출처: Chuanqi Tan et al.(2018)

대부분의 실제 응용 프로그램에서 위의 여러 기술을 조합하여 더 나은 결과를 얻는 경우가 많다. 현재 대부분의 연구는 지도 학습에 중점을 두고 있으며, 심층 신경망에 의한 비지도 또는 준 지도 학습에서 지식을 전달하는 방법은 미래에 점점 더 많은 관심을 끌 것이다. 심층 전이학습은 심층 신경망의 개발과 관련된 많은 어려운 문제를 해결하기 위해 널리 적용될 것으로 예측할 수 있다.

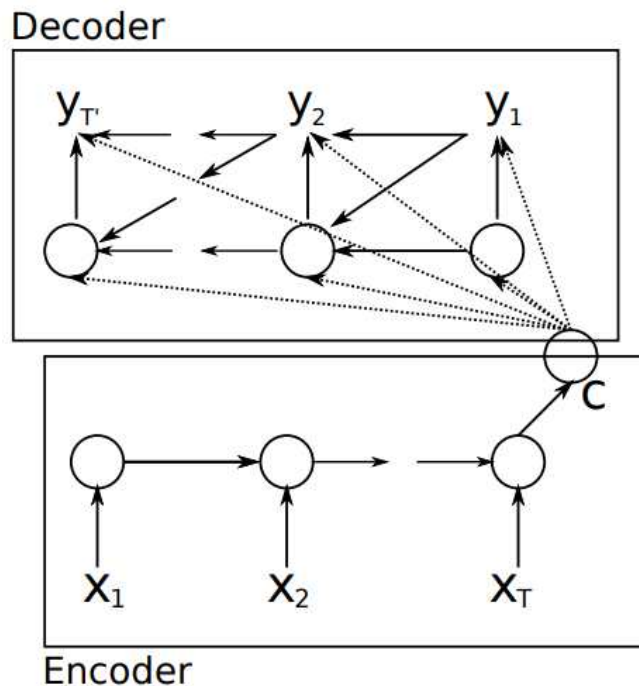
3) 어텐션 메커니즘(Attention mechanism)

자연어 처리(NLP, Natural Language Processing) 분야와 기계 번역(MT, Machine Translation) 분야에서 순환 신경망(RNN, Recurrent Neural Network)의 LSTM(Long Short-Term Memory)과 GRU(Gated Recurrent Units) 같은 모델은 대부분 최고 성능(SOTA, State-Of-The Art)을 제공하고 있다(Bahdanau et al., 2014; Cho et al., 2014; Sutskever et al., 2014).

그러나 순환신경망 모델은 순차적으로 데이터를 처리해야 하는 특성 때문에 동시에 여러 작업을 수행하지 못하여, 병렬처리의 효율도 떨어지게 된다. 이러한 순환신경망의 고질적인 문제를 해결하기 위해 Extended Neural GPU, ConvS2S, ByteNet 등과 같이 합성곱 신경망(CNN, Convolutional Neural Network)을 사용하는 경우도 있다. 하지만 합성곱 신경망을 사용하는 경우, 입력 레이어의 길이가 길어질수록 입력의 원소와 출력의 원소 사이의 의존성(Dependency)을 학습하는 것을 더 어렵게 만든다(Hochreiter et al., 2001).

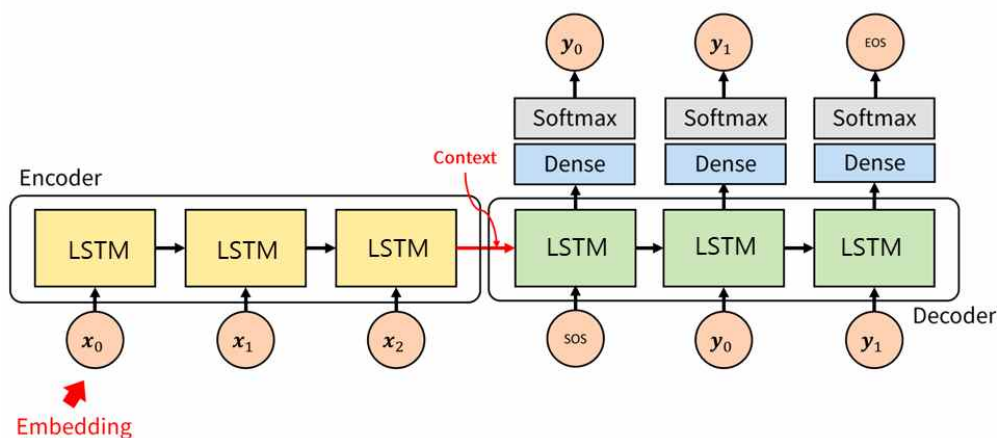
또한, 입력 문장이 긴 상황에서 자연어 처리나 번역의 품질이 떨어지는 현상이 나타났고 이러한 현상을 보정하기 위해 중요한 단어에 집중하는 어텐션 메커니즘(Attention mechanism)이 자연어 기계 번역을 위한 seq2seq(Sequence-to-sequence) 모델에 처음 도입되었다(Sutskever et al., 2014; Cho et al., 2014). 기본적인 seq2seq는 영어를 프랑스어로 번역하는 신경망 기계번역 모델을 목적으로 Cho et al.,(2014) 연구에서 제안하였고, seq2seq 모델은 내용 요약(Text Summarization), 음성 인식(Speech to Text; STT) 등에 쓰일 수 있고, 번역기에서 대표적으로 사용되는 모델이다.

기본적인 seq2seq 모델은 하나의 인코더(Encoder)와 하나의 디코더(Decoder)로 이루어져 있으며, 입력 시퀀스를 처리하는 부분을 인코더, 출력 시퀀스를 생성하는 부분을 디코더라고 한다. 인코더는 입력의 각 아 이템을 처리하여 정보를 추출한 후 하나의 벡터인 컨텍스트(Context)로 만든다. 디코더는 컨텍스트를 받아 출력할 아 이템을 하나씩 선택한다. 디 코더는 하나의 고정된 벡터로 전체의 맥락을 표현하는 방법으로, 문장 쌍 을 이루는 데이터에 대해 유용한 학습 방법으로 처음 소개되었다. 이 학 습 방법은 순환신경망의 한 종류이며, 순환신경망이 여러 개 연결된 인코 더와 디코더의 구조를 사용한다(Cho et al., 2014). <그림 II-9>와 같 이 인코더와 디코더로 불리는 두 개의 순환신경망이다.



<그림 II-9> 기본적인 seq2seq 모델, 출처: Cho et al.(2014)

Sutskever et al.,(2014) 연구에서는 순환신경망은 입력 사이즈(Input size)와 출력 사이즈(Output size)가 다른 경우에 대해서는 좋은 성능을 보일 수 없었던 한계점을 극복하기 위해 인코더 LSTM에서 하나의 컨텍스트를 생성한 뒤 디코더 LSTM에서 컨텍스트를 이용해 출력 문장(Output sentence)을 생성하는 모델을 제시했다. <그림 II-10>와 같이 입력 시퀀스를 처리하는 인코더와 출력 시퀀스를 생성하는 디코더에 LSTM 모델을 사용하였다. 이를 통해 컨텍스트가 입출력의 연관 관계가 너무 떨어져 있어서 역전파(Backpropagation) 할 때 기울기 소실(Gradient Vanishing)으로 학습이 잘되지 않는 문제를 해결할 수 있었다.

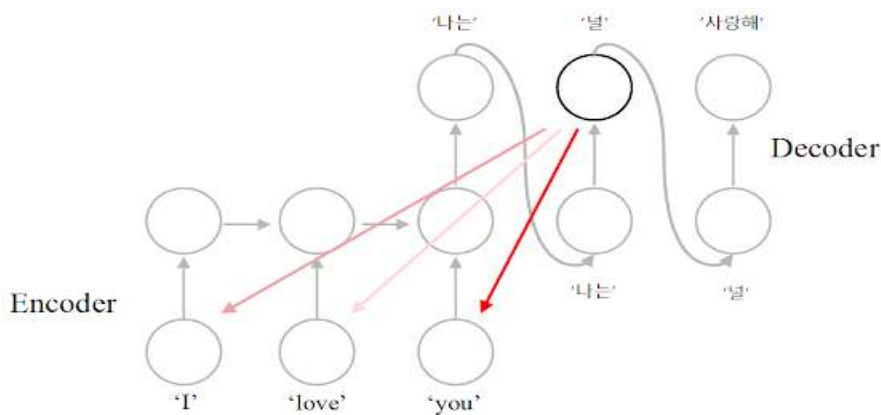


<그림 II-10> LSTM seq2seq 모델

그러나 입력 데이터 길이가 길어지면 길어질수록 Sequence-to-Sequence 학습은 인코더가 생성한 컨텍스트가 입력 데이터의 정보를 제대로 담기 어려워지는 문제가 발생한다. 긴 문장 해석과 같이 기계 번역 시 처리하기 어려운 문제를 해결하는 방법으로 제시된 것이 바로 어텐션이다(Bahdanau et al., 2014; Luong et al., 2015).

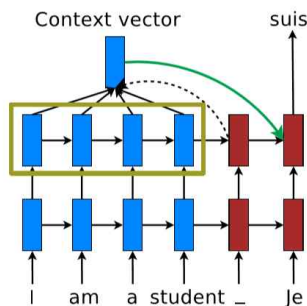
어텐션(Attention) 의미는 디코더에서 출력할 때, 어떤 인코더 정보에 집중해야 하는지 알 수 있도록 하여 출력하는 데 도움을 주겠다는 뜻이다. 이것이 어텐션 메커니즘의 기본 아이디어이다.

어텐션 메커니즘은 디코더에서 출력 단어를 예측하는 매 시점(Time-step)마다, 인코더에서의 전체 입력 문장(Sequence)을 다시 한번 참고하도록 한다. 단, 전체 입력 문장을 전부 다 동일한 비율로 참고하는 것이 아니라 해당 시점에서 예측해야 할 단어와 연관이 있는 입력 단어 부분을 좀 더 집중적으로 보게 된다. 예를 들어 <그림 II-11>과 같이 ‘너’를 예측하기 위해 ‘I’, ‘love’, ‘you’를 한 번씩 참고하면서 연관성이 있는 단어가 있는지 집중(Attention)하는 방법이다.

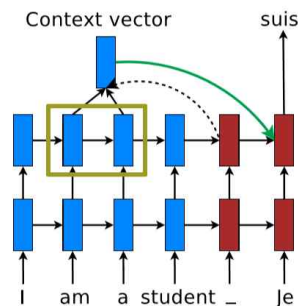


<그림 II-11> 어텐션 메커니즘의 예시

Luong et al. (2015) 연구에서는 항상 모든 소스 단어에 주의를 기울이는 글로벌 어텐션(Global attention)과 한 번에 소스 단어의 하위 집합만 보는 로컬 어텐션(Local attention)으로 어텐션 메커니즘의 두 가지 모델로 제안하였다. <그림 II-12>와 같이 글로벌 어텐션은 인코더의 전체 기억(Hidden state)에 대해서 가중치를 계산하고, 로컬 어텐션의 경우 윈도우를 이용하여 대략적으로 입력 문장의 단어를 추려서 계산한다.



Global: all source states.



Local: subset of source states.

<그림 II-12> 글로벌 어텐션과 로컬 어텐션 차이, 출처: Luong (2015)

로컬 어텐션을 사용하는 경우, 계산에 들어가는 비용이 입력 문장의 길이가 길더라도 일정하게 유지되며, 로컬 어텐션의 계산은 글로벌 어텐션보다 간단하고, 판별 가능하여 구현과 학습이 쉽다. 이것으로 기계 번역 결과와 사람이 직접 번역한 결과가 얼마나 유사한지 비교하여 번역에 대한 성능을 측정하는 방법인 BLEU(Bilingual Evaluation Understudy) 점수 기준, WMT' 14, WMT' 15의 영어에서 독일어로 번역하는 분야에서 당시의 새로운 최고 성능을 달성했다.

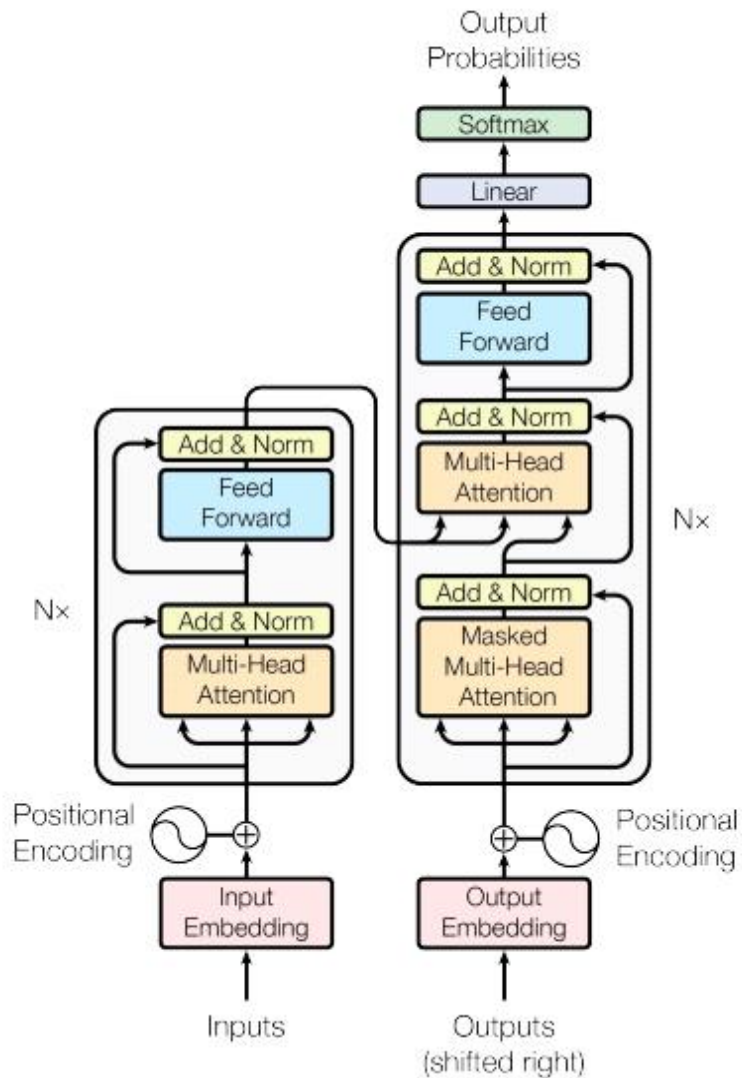
어텐션 메커니즘은 seq2seq모델이 디코딩 과정에서 현재 스텝에서 가장 관련된 입력 부분에 집중할 수 있도록 해줌으로써 기계 번역의 품질을 크게 향상시켰다. 또한, 다양한 작업에서 강력한 시퀀스 모델 및 변환 모델의 필수적인 부분이 되었으며, 입력과 출력의 원소들 사이의 거리와 무관하게 의존성을 학습할 수 있게 되었다(Bahdanau et al., 2014; Kim et al., 2017).

4) 트랜스포머(Transformer)

대부분의 어텐션은 기존의 신경망과 함께 사용됐으나 Vaswani, A., et al.(2017) 연구에서는 기존 순환신경망에서 발생하는 문제들을 해결하고 셀프 어텐션(Self-attention)을 이용하는 트랜스포머(Transformer)를 제안하였는데 이 연구에서는 반복(Recurrence)을 제거함으로써 오직 어텐션 메커니즘만을 이용해서 입력과 출력 사이의 전역 의존성(Global dependency)을 학습하게 된다. 반복 작업이 없어졌으니 당연히 더 손쉽게 병렬처리가 가능하고 훈련하는 데 훨씬 적은 시간이 소요되는 동시에 품질이 우수하다. 병렬처리를 위해 합성곱 신경망을 사용했을 때 원소들 사이의 거리가 매우 멀어지는 원거리 포지션(Position) 간의 종속성(Dependencies)을 학습하기 어려웠던 문제도 상수 개의 연산으로 고정시킬 수 있는 매우 큰 장점이 있다. 또한, 명사구나 동구와 같은 구 단위의 구조를 파악하는 구문 분석(Constituency Parsing) 분야에서 우수한 성능을 나타내며, 모델의 일반화(Generalization)도 우수하게 만들어진다.

인트라 어텐션(Intra-attention)라고도 하는 셀프 어텐션은 시퀀스의 표현을 계산하기 위해 단일 시퀀스의 서로 다른 위치를 연결하는 어텐션 메커니즘이다. 셀프 어텐션은 독해, 추상적 요약, 텍스트 수반, 학습과제 독립적인 문장 표현을 포함한 다양한 과제에서 성공적으로 사용되었다(Cheng, J., et al., 2016 ; Parikh, A. P., et al., 2016; Paulus, R., et al., 2017, Lin, Z., et al., 2017).

<그림 II-13>은 트랜스포머의 모델은 기본적으로 인코더-디코더 구조로 되어 있으며, 트랜스포머의 가장 큰 특징 중 하나는 셀프 어텐션을 수행한다는 것이다.



<그림 II-13> 트랜스포머 모델 구조, 출처: Vaswani, A., et al.(2017)

트랜스포머 모델은 기계 번역 시스템에서 악명 높은 문제인 상호 참조를 셀프 어텐션을 통해 해결하였다. <그림 II-14>와 같이 프랑스어 번역의 상호 참조 구문 예시이다.

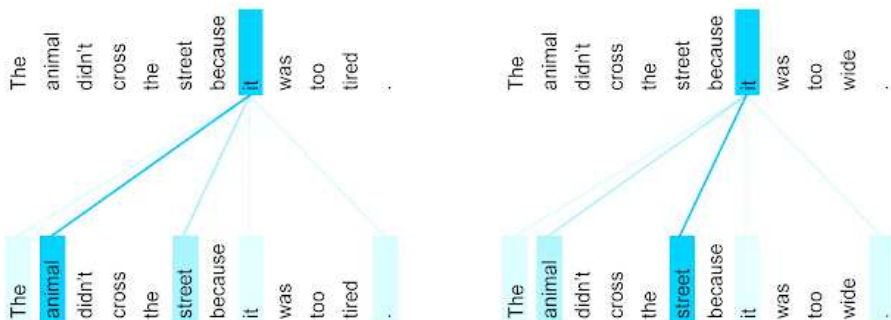
The animal didn't cross the street because it was too tired.
L'animal n'a pas traversé la rue parce qu'il était trop fatigué.

The animal didn't cross the street because it was too wide.
L'animal n'a pas traversé la rue parce qu'elle était trop large.

<그림 II-14> 상호 참조 문제 프랑스어 번역 구문 예시,

출처: <https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html>

<그림 II-14> 구문 예시에서 첫 번째 문장 쌍에서 “it”은 동물 (animal)을 나타내고 두 번째 문장에서는 거리(street)를 의미한다는 것을 대부분 사람은 분명히 알 수 있다. 다시 이 문장을 프랑스어나 독일어로 번역할 때 “it”에 대한 번역은 해당 명사의 성별에 따라 다르며 프랑스어에서는 “animal”과 “street”의 성별이 다르다. 트랜스포머는 이 두 문장을 모두 프랑스어로 올바르게 번역한다. “it”이라는 단어에 대한 최종 표현을 계산할 때 인코더가 어떤 단어에 주의를 기울였는지 <그림 II-15>와 같이 시각화하면 네트워크가 결정을 내린 방법을 알 수 있다.



<그림 II-15> 훈련된 트랜스포머의 인코더 셀프 어텐션 분포,

출처: <https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html>

<그림 II-15>는 영어에서 프랑스어로의 번역에 대해 훈련된 트랜스포머의 단어 “it”에 대한 인코더 셀프 어텐션 분포를 나타낸 것이며, 트랜스포머는 “it”이 참조할 수 있는 두 개의 명사를 명확하게 식별하고 각각의 관심 정도는 다른 컨텍스트에서 선택한 항목을 반영한다.

트랜스포머 모델은 WMT 2014 영어-독일어 번역 작업에서 28.4 BLEU 점수를 달성하여 앙상블을 포함한 기존 최상의 결과보다 2 BLEU 이상 항상 최고 성능을 달성했다.

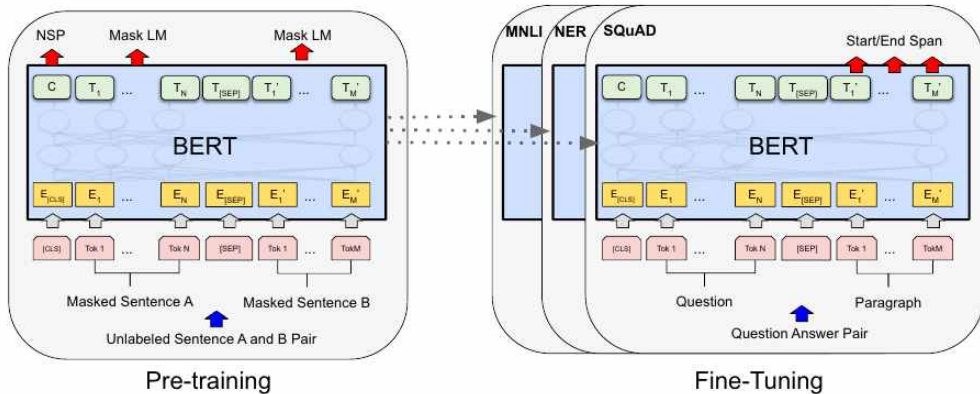
5) BERT

자연어 처리에서 “정보처리 기사”와 “운전기사”의 “기사”는 전혀 다른 의미이지만 두 가지 상황 모두 같은 벡터가 사용된다는 한계점을 사전 훈련된 언어 모델인 새로운 워드 임베딩 방법론인 ELMO 모델을 통해 사전 훈련으로 극복할 수 있었다. 자연어 처리에서 전이학습 연구가 확산하는 계기가 되어 전이학습에서 발전된 모델인 BERT가 출현하게 되었다.

BERT(Bidirectional Encoder Representations from Transformers)는 2018년에 구글이 공개한 사전 훈련된 모델이다. BERT는 트랜스포머를 이용하여 구현되었으며, 위키피디아(Wikipedia)의 25억 단어와 BooksCorpus의 8억 단어 등과 같은 레이블이 없는 텍스트 데이터로 사전 훈련된 언어 모델이다.

BERT는 모든 레이어에 양방향 문맥에서 공동으로 조절함으로써 레이블이 없는 텍스트로부터 심층 양방향 표현들을 사전 훈련하도록 디자인되었

다. 이에 관한 결과로, 사전 훈련된 BERT 모델은 작업별 구조 수정을 크게 할 필요 없이 언어 추론, 질문 답변과 같은 넓은 분야의 작업에서 하나의 출력 레이어만 추가하는 미세조정(Fine-tuning)을 통해 최고 성능(SOTA) 모델을 만들 수 있다(Devlin et al., 2018).



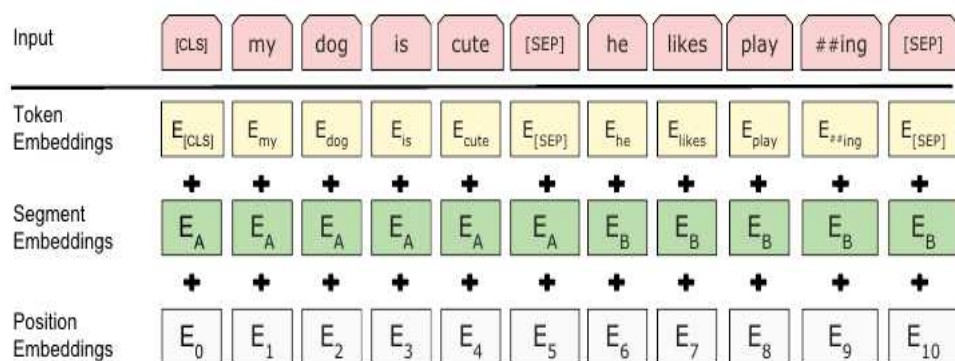
<그림 II-16> BERT 모델 학습, 출처: Devlin et al.(2018)

BERT 모델 학습은 <그림 II-16> 와 같이 사전학습(Pre-training) 단계와 미세조정(Fine-tuning) 단계로 나눌 수 있다. 사전학습 단계에서는 레이블이 지정되지 않은 데이터를 기반으로 학습을 진행한다. 미세조정 과정에서 모델은 우선적으로 전이 학습된 파라미터로 초기화된다. 이후 모델을 레이블이 지정된 데이터로 미세조정 한다. 실제 태스크(Task)에서 사용하는 모델은 초기에 동일한 파라미터로 시작하지만, 최종적으로는 서로 다른 미세조정 된 모델을 보유하게 된다. BERT는 사전학습 된 모델과 미세조정 된 모델 사이의 구조적 차이가 거의 없게 된다.

BERT 모델의 구조는 멀티 레이어 양방향 트랜스포머 인코더(Multi-layer bidirectional Transformer encoder)를 사용한다. 기존의 트랜스포머(Vaswani et al., 2017)와 거의 유사한 구조를 사용한다.

BERT에서는 두 가지 종류의 모델을 크기를 기준으로 제공한다. BERT BASE(L=12, H=768, A=12, Total Parameters=110M)와 BERT LARGE(L=24, H=1024, A=16, Total Parameters=340M)이다. 여기서 레이어(즉, 트랜스포머 블록)의 수를 L로, 숨겨진 크기를 H로, 셀프 어텐션 헤드의 수는 A로 표시한다. BERT BASE의 경우 OpenAI의 GPT와의 비교를 목적으로 동일한 크기로 제작했다. BERT의 경우는 현재 토큰의 좌/우를 모두 참조할 수 있는 양방향 셀프 어텐션을 수행하는 반면에, GPT의 경우는 현재 토큰의 왼쪽에 있는 문맥만 참조할 수 있다는 차이점이 있다.

실제 BERT 모델을 적용하기 위해서는 BERT 입력(Input) 설정과 전이 학습된 모델을 통해 최종적으로 문제를 해결하고자 하는 미세조정 단계를 거친다. 첫 번째 BERT의 입력은 <그림 II-17>과 같이 세 가지 임베딩(Token Embedding, Segment Embedding, Position Embedding)을 사용해서 데이터 입력 문장을 표현한다.

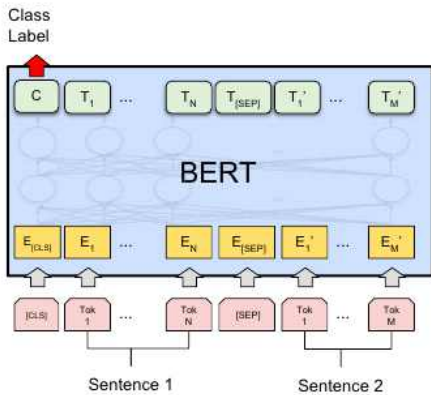


<그림 II-17> BERT 데이터 입력 표현, 출처: Devlin et al.(2018)

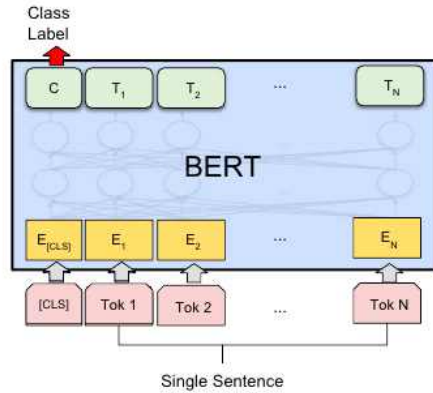
첫 번째 Token Embedding에서는 30,000개의 토큰 어휘와 함께 WordPiece 임베딩(Wu, Y., et al., 2016)을 사용한다. 또한, 특수 토큰(CLS, SEP)을 사용하여 문장을 구별하게 된다. CLS(Special Classification token)는 모든 문장의 가장 첫 번째(문장의 시작) 토큰으로 삽입된다. 이 토큰은 분류 작업(Classification task)에서는 사용되지만, 그렇지 않을 때는 무시된다. 또한 SEP(Special Separator token)를 사용하여 첫 번째 문장과 두 번째 문장을 구별한다. 여기에 Segment Embedding을 더해서 앞뒤 문장을 더욱 쉽게 구별할 수 있도록 도와준다. 이 토큰은 각 문장의 끝에 삽입된다. Position Embedding은 트랜스포머 구조에서도 사용된 방법으로 각 토큰의 위치를 알려주는 임베딩이다. 주어진 토큰에 대해 Positional, Segment, Token embedding의 값을 모두 합한 값이 그 단어의 최종 입력(Input Embedding)이 된다.

두 번째 전이 학습된 모델은 위키피디아(Wikipedia)의 25억 단어와 BooksCorpus의 8억 단어 등과 같은 레이블이 없는 텍스트 데이터로 사전 훈련된 언어 모델을 그대로 사용한다.

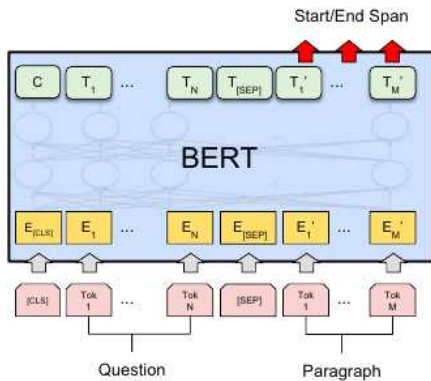
세 번째 BERT의 미세조정 단계에서는 트랜스포머의 셀프 어텐션을 이용하여, <그림 II-18>과 같이 다양한 자연어 문제에 적용하기 쉬워진다.



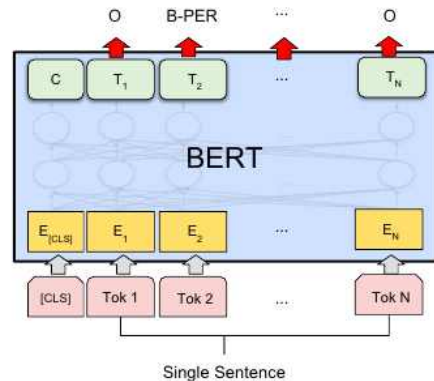
(a) Sentence Pair Classification Tasks:
MNLI, QQP, QNLI, STS-B, MRPC,
RTE, SWAG



(b) Single Sentence Classification Tasks:
SST-2, CoLA



(c) Question Answering Tasks:
SQuAD v1.1



(d) Single Sentence Tagging Tasks:
CoNLL-2003 NER

<그림 II-18> 다양한 작업(Different Tasks)에 대한
BERT 미세조정 방법, 출처: Devlin et al.(2018)

보통 자연어 문제는 단일 문자열을 입력으로 하거나 두 문자열 쌍을 입력으로 사용하는 경우가 많다. 문자열 쌍을 입력으로 해야 하는 문제의 경우 보통 문자열 쌍의 관계를 알아내기 위해 Bidirectional Cross Attention을 적용해야 하는데, 이를 적용하기 위해 입력으로 들어온 문자열 쌍을 따로 인코딩 뒤 수행하게 된다. 그러나 BERT의 셀프 어텐션은

<그림 II-19>(a)처럼 입력 문자열 쌍을 인코딩하고 양방향 어텐션(Bidirectional attention)을 적용하는 것을 한 번에 처리하는 효과 갖는다. 이를 통해 두 문자열 쌍을 입력으로 하는 문제와 한 문자열을 입력으로 하는 문제를 단일 모델, 같은 미세조정 방법으로 처리할 수 있게 되었다. 또한, CLS 토큰을 이용하여 앞 문장이 주어졌을 때 뒤의 개념을 유추하는 추론(Entailment) 문제와 텍스트에 들어있는 의견이나 감성, 평가, 태도 등의 주관적인 정보를 분석하는 감성 분석(Sentiment analysis)과 같은 분류(Classification) 문제를 풀 수 있다. CLS는 문장 내에 속하는 토큰이 아니고, 문장에 제일 앞에 위치해 모든 입력 시퀀스 전체를 바라볼 수 있는 효과를 가진다. 그리하여 미세조정 단계에서는 각각의 자연어 문제에 대해 문제에 맞는 입력과 출력을 BERT에 적용하고, 모든 사전학습된 파라미터에 end-to-end로 학습한다.

미세조정은 사전학습보다 비교적 시간이 오래 걸리지 않는다. 또한 위키 피디아(25억 단어)와 BooksCorpus(8억 단어) 문장을 기반으로 언어 표현을 이해하기 때문에 단어 간에 드러난 의미와 관계뿐만이 아니라 숨은 의미도 찾아낼 수 있다. 예를 들어 “J2EE 및 스프링 프레임워크 개발자”를 찾는 취업공고에 대해 “JAVA MVC” 역량이나 “자바 서블릿” 역량을 보유한 구직자에 대한 적합도 예측이 가능하다.

다른 BERT의 특징은 기존의 임베딩은 문장에서 단어를 순차적으로 입력받고 다음 단어를 예측하는 한 방향(uni-directional) 이지만 BERT는 문장 전체를 입력받고 단어를 예측하고 양방향(bi-directional) 학습이 가능하고 문장의 앞과 뒤의 문맥을 고려하는 것으로 이전보다 더 높은 정확도를 나타낸다. 또한, BERT의 활용은 대량의 텍스트 데이터와 다양한

언어를 적용할 수 있다는 장점 때문에, 연구자들 사이에서 가장 각광 받는 기술 중 하나이다.

최근에는 영어뿐만 아니라 100개 언어를 지원하는 ‘BERT-Base, Multilingual Cased’ 모델을 제공한다. BERT는 위키피디아를 사용하여 학습하기 때문에 BERT-Base, Multilingual Cased’ 모델에서 제공하는 언어는 가장 큰 위키피디아 정보가 있는 상위 100개 언어 기준으로 선택되었다. SKTBrain(2020) 연구에서는 한글을 대상으로 BERT-Base, Multilingual Cased 모델에 한국어를 더 학습시키는 연구가 진행되고 있다.

BERT의 이러한 특징을 활용하여 자연어 처리 분야에 BERT를 이용한 연구가 활발하게 이루어지고 있다. 홍승연 등(2020)은 BERT를 이용한 알츠하이머병 치매와 조현병 진단모델을 제안하였다. 의사와 환자 간의 대화 텍스트를 통해 알츠하이머병과 조현병 환자인지 아닌지를 분류하는 이진 분류문제로 정의하였다. 분류 모델은 BERT를 활용한 미세조정된 BERT 모델과 LSTM를 사용한 Non-BERT 모델을 비교하였다. 실험 데이터는 전북대학교 병원에서 알츠하이머병 상담 내용을 녹음하고 텍스트로 변환하여 학습 데이터를 145문서, 평가 데이터로 17 문서를 사용하였다. 조현병 또한 같은 방식으로 학습 데이터 129문서, 평가 데이터로 15 문서를 사용하였다. 실험 결과 모델 정확도(Accuracy)는 알츠하이머병 진단 BERT 모델 94.12%, Non-BERT 모델 82.54%이며, 조현병 진단 BERT 모델 86.67%, Non-BERT 모델 73.33%로 두 실험 모두 BERT를 이용한 모델이 성능이 좋았다. 의사와 환자 간의 대화 안에 표면적으로 나타나지 않고 병을 진단하는 데 필요한 내용들이 내재되어 숨은 의미

를 찾아내야 하는 부분에서 전이학습 모델인 BERT 기반 모형이 순환 신경망 기반 모형보다 정확도 성능이 높은 것으로 나타났다.

한재현 등(2020) 연구에서는 BERT 언어 모델을 사용한 언어처리 기반 질의-상담 매칭시스템을 제안하였다. 상담자의 고민 내용을 데이터로 입력받아 연애, 진로, 자존감, 대인관계 4개 카테고리 중 가장 적합한 카테고리를 분류하고, 분류된 카테고리 내의 상담자의 데이터와 입력된 상담자의 상담 내용 간에 연관성 정도를 유사도로 측정하여, 상담자에게 고민 해결에 가장 적합하다고 판단된다는 상담자를 선택할 수 있도록 3명의 상담 리스트를 제공한다. 이 연구에서는 2가지 모델을 설계하였는데 첫 번째 모델은 카테고리 분류 모델로 고민 내용을 분석하여 4가지 카테고리 분류하는 것과 두 번째 유사도 측정 모델은 카테고리 분류 후 상담자의 데이터와 상담자의 고민 내용과의 유사도를 측정하는 모델이다. 두 모델 모두 GitHub KorQQP 데이터와 Willson 상담 데이터를 기반으로 전이 학습된 BERT-Base Multilingual Cased 모델의 유사도 측정과 카테고리 분류 기능을 갖추도록 미세조정 하였다. 실험 결과 성능평가 지표인 모델 정확도(Accuracy)는 유사도 모델 75%와 분류 모델 83%를 달성하였다. 상담자의 고민 내용과 상담자의 데이터 유사도 측정 부분에서 전문가 상담사의 프로파일을 사용한 것이 아니라 비슷한 고민을 한 사람 즉 고민 내용의 유사도가 높은 사람을 상담자로 매칭 한다. 위의 연구들과 같이 BERT를 이용한 자연어 처리 분야 중에서도 상담 매칭시스템과 알츠하이머병 치매와 조현병 진단 등 환자의 상태에 관한 대화나 글 또는 상담 글에서 직접적으로 나타나지 않는 숨은 의미와 연관성을 찾는 부분에서 활발히 적용되고 있다.

<표 II-9>는 선행연구를 기반으로 BERT를 이용한 자연어 처리 분야별로 주요 연구를 정리하였다.

<표 II-9> BERT를 이용한 자연어 처리 분야 주요 연구

분야	연구자	주요 내용
질의-상담 매칭 시스템	한재현 등 (2020)	고민 내용에 맞는 상담자 매칭시스템, BERT-Base Multilingual Cased 모델의 유사도 측정과 카테고리 분류 기능을 갖추도록 미세조정
치매와 조현병 진단	홍승연 등 (2020)	알츠하이머병 치매와 조현병 진단모델, BERT를 활용한 미세조정된 BERT 모델과 LSTM를 사용한 Non-BERT 모델을 비교
상품 추천 시스템	박호연,& 김경재 (2021)	BERT 기반 감성분석을 활용한 상품 추천시스템, 사용자 리뷰와 평점을 벡터로 입력받아 BERT 감 성 분석을 할 수 있도록 구현
불안 분류기	송지수 (2020)	사회적 불안을 분석하기 위해 BERT 기반의 불안 분류기 및 사회적 불안 시각화 시스템 제시, 트위터 메시지 수집하여 학습 데이터 사용, BERT를 이용한 텍스트 분류 설계

BERT를 이용한 자연어 처리 분야 주요 연구에서는 직접적으로 나타내지 않는 숨은 의미와 연관성을 찾아 각 자연어 처리 분야에 적용하는 연구가 활발하게 진행되고 있다. 기존 연구에서는 글에 숨겨진 감정들의 의미와 질병의 연관성을 찾는 의학적인 부분에 적용되고 있다. 사회적 문제로 대두되고 있는 청년 취업과 관련된 일자리 추천과 인재추천 분야에는 아직 BERT를 적용한 특히나 양방향 추천 연구가 전무하다.

본 연구에서는 채용정보와 구직자 프로파일과 같은 문서 내 단어의 숨은 의미를 이해할 수 있는 전이학습 모형을 적용하면 기존 일자리 추천에서

사용되는 단순 워드 임베딩을 활용한 학습 모델보다 높은 성능을 나타낼 것이라는 아이디어에 착안하여 BERT 기반의 전이학습 모델을 제안하고 성능을 평가하기로 한다. 또한, 양방향 인재매칭 모델을 적용한 인재매칭 시스템으로 구축할 수 있도록 시스템 구성과 모델 연동 API 설계를 제안한다.

Ⅲ. 양방향 인재매칭 모델

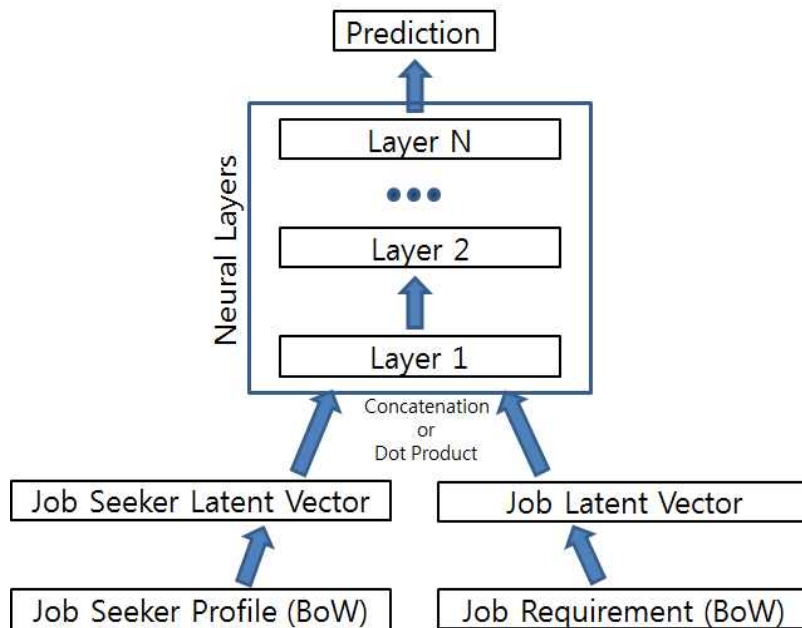
1. BERT 기반의 전이학습 모델 설계

기존 일자리 추천에서는 기업의 채용정보에서 작성된 내용에서 같은 의미지만 다른 기술용어로 작성되어 기존 워드 임베딩을 통해서는 인재추천이 어려울 뿐만 아니라, 구직자의 이력서에 작성된 내용과 연관성 있는 채용정보를 추천해 줄 수 있는 채용정보 추천이 되지 않는 단일방향 일자리 추천으로 주로 연구되었다.

본 연구는 구직자에게는 보유한 역량과 관심에 적합한 일자리를 추천할 수 있어야 하며, 기업에는 기업의 채용정보와 적합한 구직자를 추천하는 양방향 인재매칭 모델을 제안하는 것을 목표로 한다. 또한, 구직자의 이력서와 채용정보 단어의 숨은 의미와 같은 의미이지만 다른 단어로 기술된 내용의 연관성을 이해할 수 있는 사전 훈련된 언어 모델인 BERT 모델을 적용한다.

구직자의 프로파일(이력서)과 채용정보를 BertTokenizer를 이용하여 BERT 전이학습 모델에 입력하며, 최종 출력층에는 구직자 정보와 채용정보의 적합도가 출력되도록 하는 학습 구조를 제안한다. 본 연구에서는 104개 언어를 지원하는 ‘BERT-Base, Multilingual Cased’ 모델을 적용하여, 구직자 정보와 채용정보의 적합도가 출력되는 미세조정 모델(fine-tuning model)을 구성한다(Devlin et al., 2018).

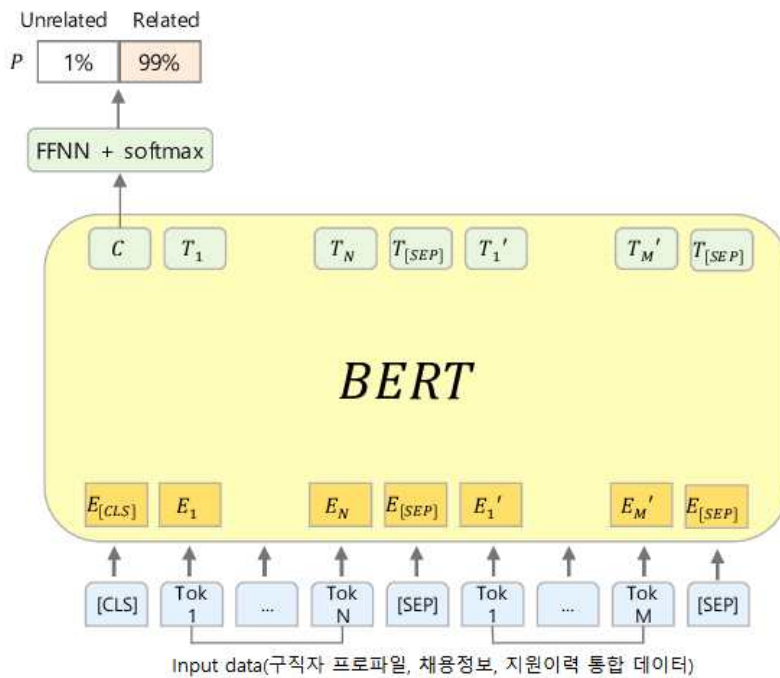
송희석(2020) 연구에서 제안한 내용 기반 척도학습모형(Content-based metric learning model for HR matching)은 <그림 III-1>과 같이 구직자 프로파일과 일자리공고 문서를 모형에 입력하기 위해 bag-of-words 벡터로 변환하는 전처리 과정을 거친다. 이렇게 입력된 구직자 프로파일과 일자리 공고문서 벡터는 각각 별도의 임베딩층을 거친 후 내적(Dot Product)을 통해 통합되어 심층 신경망을 통과하게 된다. bag-of-words 입력층 위에 임베딩층을 둔 것은 같은 구직자들이 지원한 일자리들을 유사한 일자리 벡터로 표현하거나 같은 일자리에 지원한 구직자 프로파일을 유사한 구직자 프로파일 벡터로 투사함으로써 학습 결과 입력 벡터에 시맨틱을 부여하기 위함이다. 최종 출력층은 해당 구직자 프로파일 벡터와 일자리 벡터 간 지원 이력을 나타내는 한 개의 노드로 구성되며 지원 이력이 없으면 0, 있으면 1을 출력하도록 학습이 이루어진다.



<그림 III-1> 내용 기반 척도학습모형, 출처: 송희석(2020)

내용 기반 척도학습모형은 텍스트를 bag-of-words로 변환할 때 bag-of-words의 희박도(Sparsity)가 높아 학습이 어려울 뿐 아니라 단어의 순서를 고려하지 않기 때문에 문맥적인 의미가 무시되고, 사전에 구축되지 않은 단어(OOV, Out Of Vocabulary)에 대해서는 의미적 관계를 파악하기 어렵다는 단점이 있다.

본 연구에서는 이미 훈련된 임베딩 벡터를 활용하는 미세조정 모델을 <그림 III-2>와 같이 제안한다.

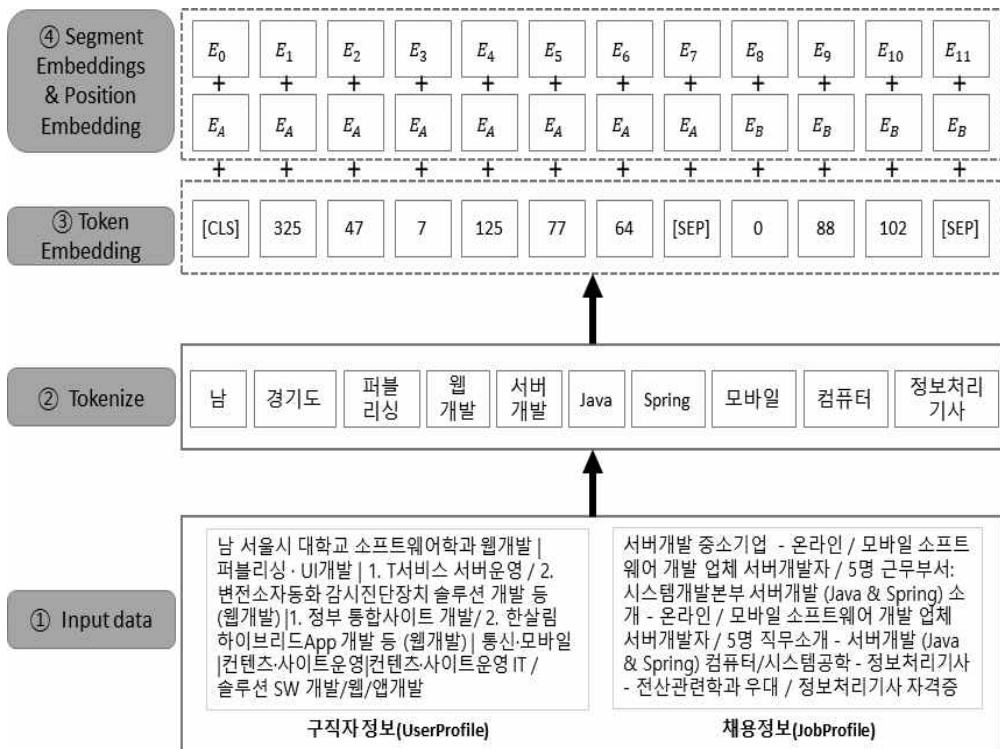


<그림 III-2> BERT 기반 양방향 인재매칭 모델

104개 언어를 지원하는 ‘BERT-Base, Multilingual Cased’ 모델을 적용하여, 구직자 정보와 채용정보의 적합도가 최종 출력되는 미세조정 모델 (fine-tuning model)을 설계하였다.

1) BERT 모델 입력

BERT 모델을 이용한 양방향 인재매칭 모델은 <그림 III-3>과 같이 구직자의 프로파일(이력서)과 채용정보를 BERT 전이학습 모델에 반영할 수 있도록 Tokenize, Token Embeddings, Segment Embeddings & Position Embeddings 과 같은 입력 과정을 거친다.



<그림 III-3> BERT 기반 양방향 인재매칭 모델 입력

구직자 프로파일 프로파일은 이름, 생년월일, 성별, 주소, 대학교, 학력, 전공, 경력, 경력 사항, 자격증, 연봉, 이력서 등 총 15개의 칼럼으로 이루어져 있으며, 전체 데이터 수는 6,028건이다. <그림 III-4>는 구직자 프로파일을 정보를 요약한 것이다.

```

RangeIndex: 6028 entries, 0 to 6027
Data columns (total 15 columns):
#   Column                Non-Null Count  Dtype
---  ---
0   user_id                6028 non-null   int64
1   name                   6028 non-null   object
2   birth                  6028 non-null   int64
3   sex                    6025 non-null   object
4   addr                   6026 non-null   object
5   univ                   6026 non-null   object
6   education               6026 non-null   object
7   major                  5824 non-null   object
8   work_period            5915 non-null   object
9   work_company           6026 non-null   object
10  skill                   5728 non-null   object
11  pay                     4367 non-null   object
12  profession              5973 non-null   object
13  profession_detail       5938 non-null   object
14  resume                  6028 non-null   object
dtypes: int64(2), object(13)
memory usage: 706.5+ KB

```

<그림 III-4> 구직자 프로파일

채용정보 프로파일은 채용직무, 기업형태, 회사소개, 직무소개, 자격요건, 우대사항으로 이루어져 있으며, 전체 데이터 수는 352건이다. <그림 III-5>는 채용정보 프로파일을 간략하게 정리한 것이다.

```

RangeIndex: 352 entries, 0 to 351
Data columns (total 7 columns):
#   Column                Non-Null Count  Dtype
---  ---
0   order_id               352 non-null   int64
1   job_name               352 non-null   object
2   company_size           352 non-null   object
3   company_overview       314 non-null   object
4   job_desc               329 non-null   object
5   job_requirement        138 non-null   object
6   etc                    315 non-null   object
dtypes: int64(1), object(6)
memory usage: 19.4+ KB

```

<그림 III-5> 채용정보 프로파일

지원 이력 데이터는 구직자 아이디, 채용정보 아이디, 지원 이력으로 이루어져 있으며, 전체 데이터 수는 60,362건이다. <그림 III-6>은 지원 이력을 간략하게 정리한 것이다.

```
RangeIndex: 60362 entries, 0 to 60361
Data columns (total 3 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   user_id     60362 non-null  int64
1   order_id    60362 non-null  int64
2   rating      60362 non-null  int64
dtypes: int64(3)
memory usage: 1.4 MB
```

<그림 III-6> 지원 이력 데이터

구직자 프로파일과 채용정보는 각각 구직자 프로파일(UserProfile)과 채용정보 프로파일(JobProfile)을 칼럼으로 변환 후 지원 이력(Label)을 포함하여 하나의 데이터로 통합한 통합 데이터를 생성한다. 이렇게 통합된 데이터는 총 59,694건이며, BertTokenizer를 이용하여 텍스트를 더 이상 나눌 수 없는 단어 형태(Token)로 분리한다. <그림 III-7>은 통합 데이터 정보를 간략하게 정리한 것이다.

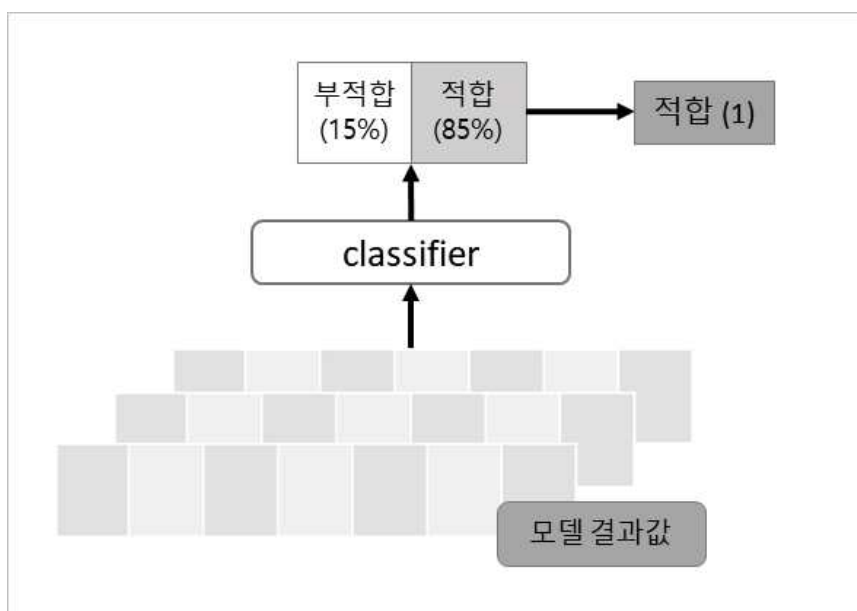
```
Int64Index: 59694 entries, 0 to 59693
Data columns (total 5 columns):
#   Column          Non-Null Count  Dtype
---  ---
0   user_id         59694 non-null  int64
1   order_id        59694 non-null  int64
2   rating          59694 non-null  int64
3   userProfile     59694 non-null  object
4   jobProfile      59694 non-null  object
dtypes: int64(3), object(2)
memory usage: 2.7+ MB
```

<그림 III-7> 통합 데이터

분리한 단어들은 Token Embeddings을 통해 단어의 고윳값(ID)을 부여하고 Segment Embeddings & Position Embeddings 통해 문장의 순서와 위치값을 설정한다. 이러한 전처리 과정을 거친 입력 데이터를 BERT 모델의 최종 입력(Input Embedding)으로 한다.

2) BERT 모델 미세조정

최종 출력층을 추가하는 미세조정 단계에서는 <그림 III-8>과 같이 BERT 전이학습 모델에 BERT 모델 입력을 통해 추가 학습한 모델 결과값을 BERT 모델의 입력 (구직자 프로파일-채용정보)에 대한 지원 이력 등의 상호작용 내역을 반영한 한 개의 노드로 구성하여 적합도 점수에 따라 매칭이 부적합하면 0 적합하면 1을 출력하도록 분류 모델로 미세조정하였다.



<그림 III-8> BERT 기반 양방향 인재매칭 모델 미세조정

3) BERT 모델 하이퍼 파라미터 튜닝

하이퍼 파라미터(Hyper parameter)는 머신러닝 및 딥러닝 모델의 입력값으로 해당 모델이 목표 데이터 특성으로부터 일반화된 추론 성능을 훈련할 수 있도록 제어하는 기능을 수행한다. 이러한 하이퍼 파라미터 튜닝은 인공 신경망 훈련 시 가장 우수한 성능을 도출할 수 있는 하이퍼 파라미터를 찾아내는 기술을 의미하며, 하이퍼 파라미터로는 옵티마이저(Optimizer), 학습률(Learning Rate), 학습률 스케줄링 방법, 손실함수(Loss Function), 훈련 반복횟수, 가중치 초기화 방법, 정규화 방법, 적층할 계층의 수 등이 고려될 수 있다(이용주 등, 2019; Choi et al., 2019; Ruder, 2016). <표 III-1>은 하이퍼 파라미터 튜닝 기법을 정리한 자료이다.

<표 III-1> 하이퍼 파라미터 튜닝 기법

종류	설명	적용 시 고려사항
옵티마이저 (Optimizer)	손실함수를 통해 구한 차이를 사용해 기울기(gradient)를 구하고 가중치를 학습에 어떻게 반영할 것인지를 결정하는 방법	<ul style="list-style-type: none"> ◦ 최적의 알고리즘 (Adam, RAdam, SGD, GD등) 고려
학습률 (Learning Rate)	기울기의 방향으로 얼마나 빠르게 이동할 것인지 결정하는 변수	<ul style="list-style-type: none"> ◦ 너무 작으면 학습 속도 늦음 ◦ 너무 크면 학습 불가
손실 함수 (Cost Function)	입력에 따른 기대 값과 실제 값의 차이를 계산하는 함수	<ul style="list-style-type: none"> ◦ 평균 제곱 오차 ◦ 교차 엔트로피 오차

정규화(일반화) 파라미터 (Regularization parameter)	과적합(Overfitting) 문제 회피 위해 L1 또는 L2 정규화 방법 사용	◦ 사용하는 일반화 변수도 하이퍼 파라미터로 분류
미니 배치 크기 (Mini-batch Size)	배치셋 수행을 위해 전체 학습 데이터를 등분하는 (나누는) 크기	◦ 가용 메모리 크기 및 epoch 수행 성능을 고려
훈련 반복 횟수 (Training Loop)	학습의 조기 종료를 결정하는 변수	◦ 학습 효율이 떨어 지는 시점을 적절 히 판단
은닉층의 뉴런 개수 (Hidden Unit)	훈련 데이터에 대한 학습 최적화 결정 변수	◦ 첫 Hidden Layer 의 뉴런 수가 Input Layer 보다 큰 것이 효과적
가중치 초기화 (Weight Initialization)	학습 성능에 대한 결정 변수	◦ 모든 초기값이 0 일 경우 모든 뉴런 이 동일한 결과

출처: Choi et al.(2019)

본 연구의 학습 모델의 하이퍼 파라미터(Hyper parameter) 설정은 학습률, 옵티마이저, 손실함수, 훈련 반복횟수, 가중치를 설정하였다. 옵티마이저는 RectifiedAdam(RAdam)을 사용하였고, 학습률은 $1.0e-5$ 설정하였다. 손실함수는 이진 분류문제(적합/부적합)로 정의하였기 때문에 BinaryCrossentropy를 사용하였다, 과대 적합(Overfitting) 문제를 해결하기 위해서 가중치(Weight decay)를 0.0025 설정하였다. 마지막으로 모델 훈련 반복횟수는 총 4번(Epochs=4)을 진행하였다.

모델 구현과 실행은 python 3.7, Google Colab의 TPU를 사용하였다.

<표 III-2>는 본 연구에 적용한 미세조정 및 하이퍼 파라미터를 정리한 것이다

<표 III-2> BERT 미세조정 및 하이퍼 파라미터

Parameter	Base
학습률	$1.0e-5$
옵티마이저	RectifiedAdam (RAdam)
손실함수	BinaryCrossentropy
훈련 반복횟수	epochs=4
가중치	0.0025
모델 구현	python 3.7, Google Colab의 TPU

IV. 실험 및 분석결과

1. 데이터 셋

실험에 사용한 데이터는 국내 헤드헌터 업체에 축적된 데이터로 6,028명의 구직자 이력서와 352개의 채용정보, 그리고 지난 수년간 이들 구직자가 특정 채용정보에 지원한 이력 데이터 60,362건으로 구성되어 있다. 지원 이력 데이터는 긍정사례 17,995건과 부정사례 42,367건으로 긍정사례 29.8%와 부정사례 70.2%의 비율로 구성되었다. 구직자 이력서는 성별, 소재지, 대학교, 최종학력, 전공, 경력연수, 근무회사, 전문기술, 희망 연봉, 전문경력, 경력 상세정보와 장문의 자기소개서 정보를 텍스트 형태로 포함하고 있다. 채용정보는 헤드헌터가 기업으로부터 의뢰받은 정보로 채용직무, 기업형태, 회사소개, 직무소개, 자격요건, 우대사항 등의 텍스트 형태이다. 마지막으로 지원 이력 정보는 구직자 ID, 채용정보 ID, 지원 여부(1; 지원, 0; 지원 안함)로 구성된 데이터이다. 실험 데이터는 교차 검증을 위해 훈련용 데이터는 전체 데이터의 80%, 시험용 데이터는 전체 데이터의 20%로 임의 추출하여 실험을 진행하였다.

<표 IV-1> 실험 데이터 셋

데이터 셋	데이터 수
구직자 프로파일	6,028 건
채용정보	352 건
지원 이력 데이터	전체 60,362 건, 긍정사례 17,995건(29.8%), 부정사례 42,367건(70.2%)
훈련용 데이터	47,755건(전체 80%)
시험용 데이터	11,939건(전체 20%)

2. 성능평가 방법

본 연구에서는 성능평가 지표를 제시하여 성능평가를 수행한다. 추천 문제를 정의하는 방법은 특정 구직자가 특정 채용정보에 적합한지 부적합한지에 대한 여부를 예측하는 방법으로, 추천 문제를 0 또는 1의 이진 분류문제로 정의한다. 여기서 이진 분류문제의 종속변수 즉, 레이블(Label)은 지원 여부로 1은 채용정보에 적합 또는 구직자의 적합한 것으로 긍정을 나타내며, 0은 채용정보에 적합 또는 구직자의 적합하지 않으면 부정을 나타내도록 코딩하였다. 성능평가를 위한 지표로 정밀도(Precision)와 재현율(Recall), f1-score, AUC(Area Under Curve)를 사용하였다. 본 연구에서 정밀도는 추천모형에서 긍정(‘1’)이라고 예측된 구직자-채용정보 쌍 중 실제 지원한 적이 있던 구직자-채용정보 쌍의 비율로 계산한다. 재현율은 실제 지원한 적이 있던 구직자-채용정보 쌍 중 추천모형에서도 긍정(‘1’)으로 예측된 구직자-채용정보 쌍의 비율로 계산한다.

재현율과 정밀도 사이는 트레이드 오프(Trade-off) 라는 관계가 존재한다. 즉, 정밀도와 재현율을 둘 다 모두 최댓값으로 할 수가 없다는 것으로 정밀도가 극도로 높게 된다면 재현율은 낮아질 수밖에 없고, 반대로 재현율이 극도로 높게 된다면 정밀도는 낮아질 수밖에 없다. 재현율과 정밀도는 상호 보완적인 평가 지표이기 때문에 f1-score를 평가 지표에 추가하였다. f1-score는 정밀도와 재현율을 이용하여 조화평균(Harmonic mean)을 이용한 값으로, 주로 분류 클래스 간 데이터가 심각한 불균형(Imbalanced data)을 이룰 때 사용한다.

<표 IV-2>는 성능지표 중 정밀도의 계산 공식이다.

<표 IV-2> 정밀도 계산 공식

$$\text{정밀도(Precision)} = \frac{\|P \cap A\|}{\|P\|}$$

$\|P \cap A\|$: 긍정('1')으로 예측된 구직자-채용정보 쌍 중
실제 지원이력이 있는 구직자-채용정보 쌍의 수

$\|P\|$: 긍정('1')으로 예측된 구직자-채용정보 쌍의 수

<표 IV-3>은 성능지표 중 재현율의 계산 공식이다.

<표 IV-3> 재현율 계산 공식

$$\text{재현율(Recall)} = \frac{\|A \cap P\|}{\|A\|}$$

$\|A \cap P\|$: 실제 지원이력이 있는 구직자-채용정보 쌍 중
추천모형에서도 긍정('1')으로 예측된 구직자-채용정보 쌍의 수

$\|A\|$: 실제 지원이력이 있는 구직자-채용정보 쌍의 수

<표 IV-4>는 성능지표 중 f1-score의 계산 공식이다.

<표 IV-4> f1-score 계산 공식

$$f1\text{-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

AUC(Area Under Curve) 값은 ROC(Receiver Operating Characteristic) 그래프의 면적으로 계산하며 최댓값은 1이 된다. ROC 그래프는 모든 임계값에서 분류 모델의 성능을 보여주는 값으로 이진 분류문제의 경우 분류 임계 값(Classification threshold)에 따라 예측 정확도가 달라질 수 있다. 따라서 이진 분류에서 분류할 기준이 되는 임계

값을 낮게 설정하면 더 많은 사례가 양성으로 분류되기 때문에 참 양성 비율(TPR, True Positive Rate)과 허위 양성 비율(FPR, False Positive Rate)이 모두 높아지게 된다. AUC는 분류에 사용되는 변수 또는 모형의 성능을 평가하기 위해 가장 널리 사용되는 척도이다.

이외에도 실제 추천 시스템에 적용하기 위해서는 특정 사용자에게 가장 적합할 것으로 예상하는 N개의 채용정보를 추천하는 Top-N 추천이 중요하다. 본 연구에서는 Top-N 추천의 성능을 평가하기 위해 추천하고자 하는 Top-N 추천 재현율(Recall and Top-N; REC@N)과 Top-N 추천 정확도(Average Accuracy Score and Top-N; AAS@N)의 두 가지 성능평가 지표를 사용하였다(Song, 2020). Top-N 추천 재현율(Recall and Top-N; REC@N)은 과거 사용자가 지원 이력이 있는 채용정보가 Top-N 추천 리스트에 포함되어 있는지를 계산하는 것으로, 사용자별로 과거에 지원한 채용정보가 Top-N 추천 리스트에 포함되어 있으면 1, 없으면 0으로 계산하여 전체 사용자에게 대해 평균한 값으로 계산된다. <표 IV-5>는 Top-N 추천 재현율(REC@N, Recall and Top-N)의 계산 공식이다.

<표 IV-5> Top-N 추천 재현율(REC@N, Recall and Top-N)

$$REC@N = \frac{1}{m} \sum_{i=1}^m \Pi_i^{j_i}, \Pi_i^{j_i} = \begin{cases} 1, & \text{if } j_i \in TopNlist \text{ for } i \\ 0, & \text{else} \end{cases}$$

where i : user, j_i : Job post j to which user i had been applied

Top-N 추천 정확도(AAS@N, Average Accuracy Score and Top-N)는 더욱 정밀하게 Top-N 추천의 정확도를 평가하기 위한 지표로, 과거 구직자가 지원 이력이 있는 채용정보가 Top-N 추천 리스트에 몇 번째로 나타났는지를 고려한 지표이다. 특정 구직자가 지원한 바 있던 채용정보가 Top-N 추천 리스트의 맨 처음에 나타나면 1을, k번째 나타나면 $(N-(k-1))/N$ 을, 추천 리스트에 없으면 0을 점수로 부여하는 방법으로 다음과 같은 식으로 계산한다. <표 IV-7>은 Top-N 추천 정확도(AAS@N, Average Accuracy Score and Top-N)의 계산 공식이다.

<표 IV-7> Top-N 추천 정확도

(AAS@N, Average Accuracy Score and Top-N)

$$AAS@N = \frac{1}{m} \sum_{i=1}^m X_i^{j_i}, X_i^{j_i} = \frac{N - (rank_{j_i} - 1)}{N}$$

where $rank_{j_i}$: rank of j_i in Top N list for i

3. 실험 결과

본 연구에서는 구직자 이력서와 채용정보 문서의 텍스트를 bag-of-words 벡터로 변환하여 동시에 입력한 후 심층 신경망을 거쳐 최종 출력층에서 적합도 여부를 학습하는 내용 기반 척도학습모형(송희석, 2020)을 양방향 인재추천 모형의 비교분석 대상 모형으로 선정하여 양방향 추천 모형의 성능평가를 시행하기로 한다.

실험에 사용된 첫 번째 모형은 일자리공고 벡터와 구직자 프로파일 벡터를 결합(concatenation)방식에 의해 통합하는 내용 기반 척도학습 모형(CBML)이다. 두 번째 모형은 BERT 모형을 이용한 양방향 인재매칭 모형(BERT_HR)이다. 내용 기반 척도학습 모형(CBML)은 채용정보 안에서 단어 사전을 만드는 방식인 bag-of-words를 사용하여 상위 2000개의 어휘를 사용하여 학습에 사용하였고, BERT 모형을 이용한 양방향 인재매칭 모형(BERT_HR)은 구직자 프로파일 6,028건 채용정보 352건 지원한 이력 데이터 60,362건의 데이터를 BERT 사전학습 모델에 임베딩하기 위해 userProfile(구직자 프로파일), jobProfile(채용정보), label(지원 이력)을 하나의 파일로 통합하는 전처리 과정을 통하여 59,694건의 통합 데이터를 구성하였다. 훈련용 데이터는 전처리를 통한 통합 데이터의 80%, 시험용 데이터는 통합 데이터의 20%로 임의 추출하여, 통합 데이터 59,694건 중 훈련용 데이터 47,755건, 시험용 데이터 11,939건을 실험에 사용하였다.

모델 학습에는 한국어 포함 104개 언어를 지원하는 ‘BERT-Base, Multilingual Cased’ 전이 학습된 모델을 사용하였고, 임의로 추출된 훈

련용 47,755건의 데이터를 BertTokenizer를 이용하여 BERT 전이학습 모델에 임베딩 할 수 있는 입력층을 통해 추가 학습에 사용하였다.

제안 모델의 하이퍼 파라미터(Hyper parameter) 설정은 다음과 같다. 먼저 모델을 학습할 때 데이터의 실제 결과와 모델이 예측한 결과를 기반으로 잘 줄일 수 있게 만들어주는 역할을 하는 옵티마이저(Optimizer)는 RectifiedAdam을 설정하고, 그에 따른 학습률은 $1.0e-5$, 과대 적합(Overfitting) 문제를 해결하기 위해서 Weight decay를 0.0025 설정하였다. 또한 손실함수(Loss Function)는 이진 분류문제 정의에서 가장 성능이 좋은 BinaryCrossentropy를 사용하고 모델 훈련은 총 4번(Epochs=4)을 진행하였다.

교차 검증(Cross Validation)을 위해 학습에 사용하는 훈련용 데이터와 학습에 사용되지 않은 시험용 데이터로 분리하여, 시험용 데이터인 11,939건을 성능평가에 사용하였다. 추천 성능평가 지표는 정밀도, 재현율, f1-score, AUC를 측정하였다. 실험 결과 모델별 정밀도와 재현율, f1-score, AUC를 <표 IV-7>과 <그림 IV-7>로 정리하였다.

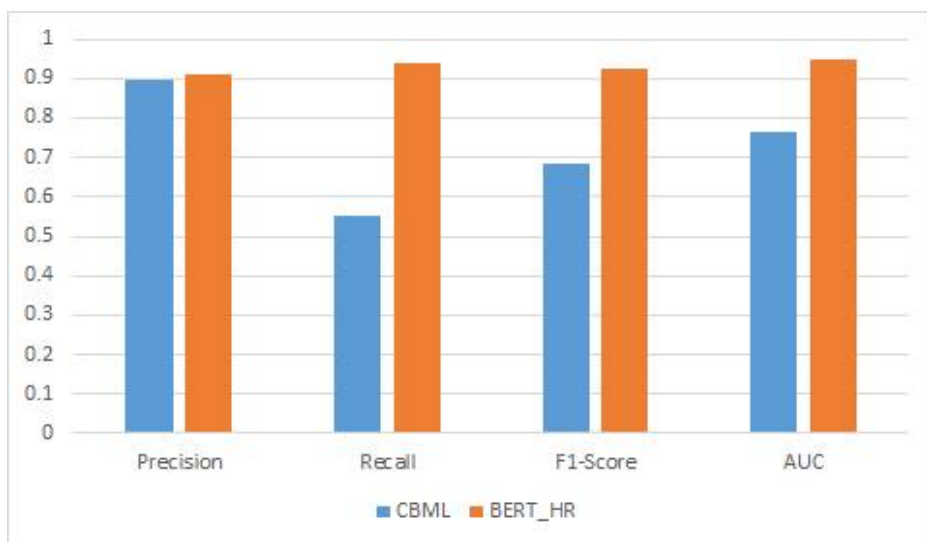
<표 IV-7> 정밀도, 재현율, f1-score, AUC 모델 성능 비교

모델	정밀도	재현율	f1-score	AUC
BERT_HR	0.9104	0.9371	0.9236	0.9487
CBML	0.8961	0.5514	0.6827	0.7628

<표 IV-1>과 같이 정밀도와 재현율, f1-score, AUC 측면에서 내용 기반 척도모형(CBML)이 정밀도는 0.8961로 높지만, 재현율은 0.5514로 낮게 나타났다. BERT 인재매칭 모델(BERT_HR)은 정밀도, 재현율,

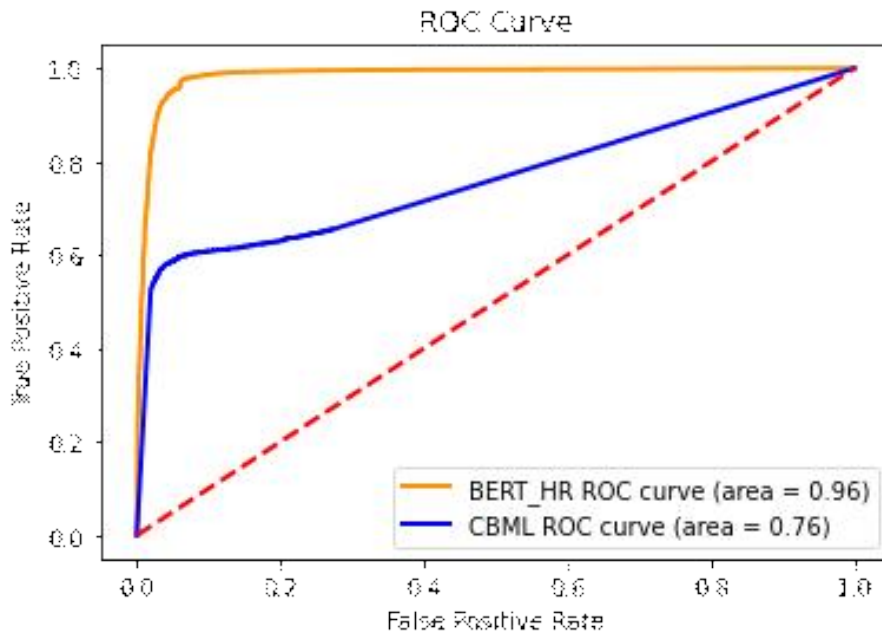
f1-score, AUC 모두 0.9 이상을 나타내며, 내용 기반 척도모형(CBML)보다 모든 측면에서 높게 나타났다. 구직자에게 적합한 모든 채용정보가 추천되었는지를 확인하는 지표인 재현율은 BERT 인재매칭 모델(BERT_HR)이 내용 기반 척도모형(CBML)에 비해 아주 높은 성능을 나타냈다. 이는 내용 기반 척도모형(CBML)의 경우 bag-of-words를 사용하기 때문에 학습되지 않은 실험 데이터의 사전에 구축되지 않은 단어(OOV; Out Of Vocabulary) 문제를 해결하기 어렵지만, BERT 인재매칭 모델(BERT_HR)은 이미 전이 학습된 모델을 적용했기 때문에 새로운 데이터에 대해서 구직자 이력서와 채용정보 사이에 다양한 전문 기술용어 간의 의미적 관계를 파악할 수 있었기 때문으로 해석된다.

실제 인재매칭 추천 시스템에서는 채용정보가 구직자에게 적합한 채용정보 인지를 확인하는 지표인 정밀도를 높게 유지하는 것이 중요하다. 채용정보와 구직자 추천 문제에서 재현율보다 정밀도가 더욱 중요한 지표라는 사실을 고려하더라도 <그림 IV-1>과 같이 두 가지 모델 중 BERT 인재매칭 모델(BERT_HR)이 모든 성능지표에서 가장 높은 성능을 나타냈으며, BERT 인재매칭 모델(BERT_HR)이 인재매칭 시스템 실제 적용이 가능할 것으로 판단된다.



<그림 IV-1> 정밀도, 재현율, f1-score, AUC 성능 비교

<그림 IV-2>는 두 모델의 ROC 그래프를 나타내고 있다. ROC 그래프의 곡선(ROC Curve)은 이진 분류에서 분류할 기준이 되는 모든 임계 값(Threshold)에서 분류 모델의 성능을 보여주는 것이다. 이진 분류 모델에서 두 클래스(0:부적합; 1:적합)를 더 잘 구별할 수 있을 때 ROC 곡선은 그래프의 왼쪽 위 끝에 더 가까워지게 된다.



<그림 IV-2> ROC 곡선 그래프 성능 비교

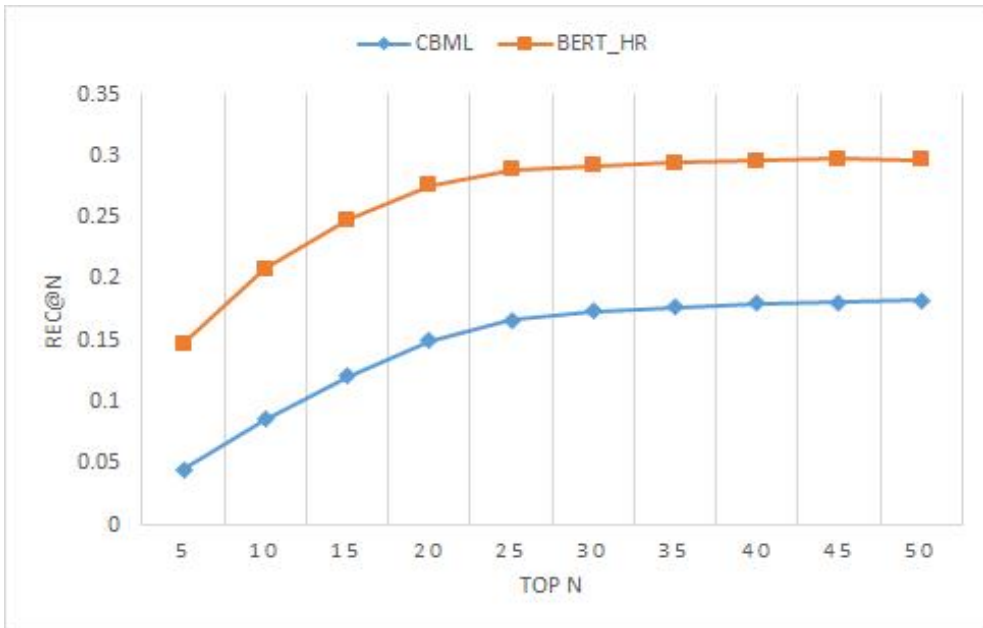
BERT 인제매칭 모델(BERT_HR)이 내용 기반 척도모형(CBML)보다 ROC 곡선이 그래프의 왼쪽 위 끝에 더 가깝게 그려지고 있어서 더 좋은 분류 모델이라고 해석할 수 있다. 또한, ROC 곡선 아래쪽의 면적을 나타내는 AUC가 높다는 것은 클래스를 구별하는 모델의 성능이 훌륭하다는 것을 의미한다.

다음으로, 모델별로 Top-N 추천 개수를 최소 5개에서 50개까지 변화시키면서 Top-N 추천 재현율과 Top-N 추천 정확도를 통해 추천 성능을 비교하여 <표 IV-8>로 정리하였다.

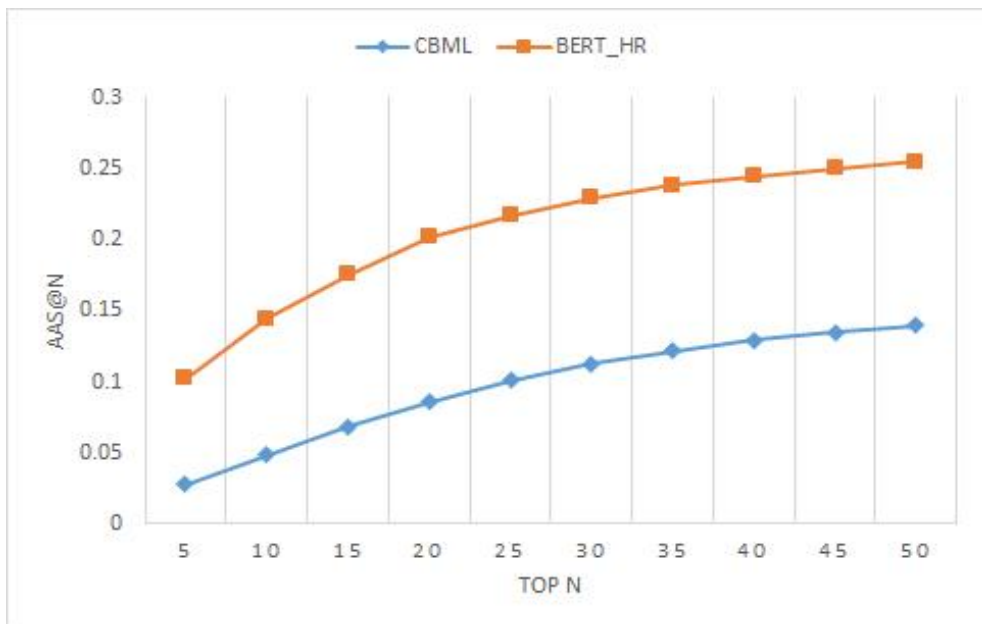
<표 IV-8> Top-N 추천 재현율과 Top-N 추천 정확도 추천 성능 비교

Top N	Top-N 추천 재현율 (REC@N)		Top-N 추천 정확도 (AAS@N)	
	BERT_HR	CBML	BERT_HR	CBML
5	0.1472	0.045	0.1022	0.0275
10	0.2085	0.0859	0.1442	0.0488
15	0.2481	0.121	0.1754	0.0682
20	0.2759	0.1495	0.2018	0.0861
25	0.289	0.1664	0.2166	0.101
30	0.2923	0.1742	0.2291	0.1128
35	0.2944	0.1773	0.2383	0.1219
40	0.2963	0.1802	0.2443	0.129
45	0.2974	0.1811	0.2501	0.1348
50	0.2973	0.1827	0.2547	0.1395

<그림 IV-3>은 BERT 인재매칭 모델(BERT_HR), 내용 기반 척도모형(CBML)의 Top-N 추천 재현율(REC@N)과 <그림 IV-4>는 Top-N 추천 정확도(AAS@N)를 각각 나타내고 있다. 실험 결과 Top-N 추천 재현율과 Top-N 추천 정확도 모두 BERT 인재매칭 모델(BERT_HR)이 내용 기반 척도모형(CBML) 보다 높은 성능을 나타내고 있다. 그리고 Top-N 개수에 따라 성능 차이가 나타나지만, Top 25개 이후 비교적 일정한 성능을 보인다. 따라서 실제 인재매칭 추천 시스템에서의 Top N 기본값을 25 값으로 설정하는 것이 효과적일 것으로 예상된다.



<그림 IV-3> Top-N 추천 재현율(REC@N) 성능 비교



<그림 IV-4> Top-N 추천 정확도(AAS@N) 성능 비교

V. 양방향 인재매칭 시스템 제안

1. 양방향 인재매칭 시스템 설계

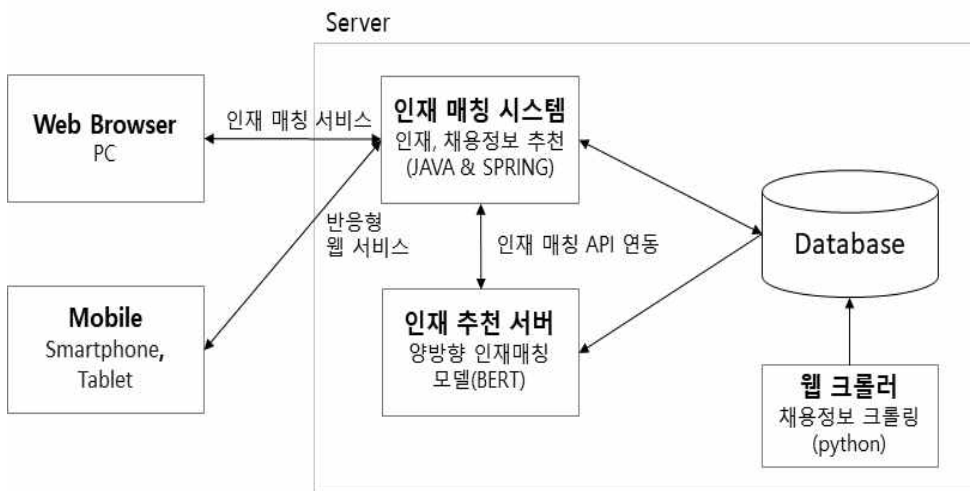
BERT를 적용한 추천 모델의 설계와 성능평가를 통해 활발한 연구가 이루어지고 있다. 그러나 기존 연구에서 성능평가까지 완료된 모델을 실제 시스템으로 구축하거나, 시스템 구성을 제안하는 사례는 드물다. BERT를 적용한 모델의 경우 기존 웹 시스템 구축에서 사용되는 일반적인 프로그래밍 언어(java, php, asp 등)와 다른 프로그래밍 언어인 파이썬(Python) 또는 파이토치(PyTorch)를 사용하여 개발한다. 서로 다른 프로그래밍 언어를 사용하기 때문에 웹 서비스를 위한 시스템과 추천 모델과 연동하기 위한 시스템 설계는 필수적이다.

본 연구에서 기업에는 적합한 구직자를 추천하고, 구직자에게는 보유한 역량과 관심에 적합한 채용정보 추천하는 양방향 추천을 웹서비스로 제공할 수 있도록 BERT 기반의 양방향 인재매칭 추천 모델과 연동하는 양방향 인재매칭 시스템 설계 및 구성을 제안하는 것을 목표로한다.

1) 양방향 인재매칭 시스템 구성

본 연구에서 제안하는 양방향 인재매칭 시스템 구성도는 <그림 V-1>과 같이 사용자(User)와 웹 브라우저(Web Browser) 또는 모바일 디바이스로 인재매칭 시스템의 인재매칭 서비스를 이용할 수 있다. 인재매칭 시스템은 인내 추천 서버와 API 연동을 통해 인재추천 서비스를 제공한다. 채용정보는 웹크롤러를 통해 주기적으로 최신의 채용정보를 데이터베

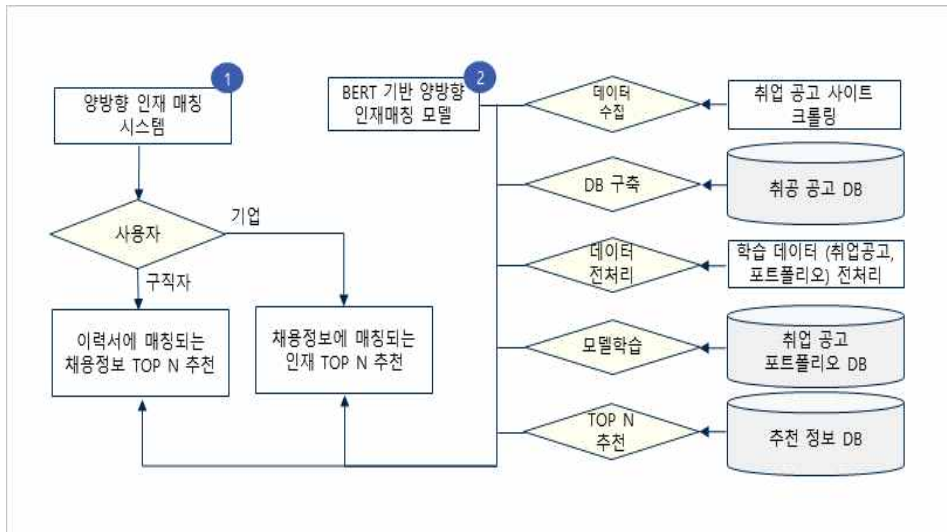
이스에 저장하도록 한다. 이를 통해 구직 희망자는 인재매칭 시스템을 통해 최신 채용정보를 확인할 수 있다. 인재매칭 시스템과 인재추천 서버는 데이터베이스를 공유하여, 최신 채용정보 및 사용자 포트폴리오에 매칭되는 인재와 채용정보를 실시간 추천이 가능하도록 양방향 인재매칭 시스템 구성도를 설계하였다.



<그림 V-1> 양방향 인재매칭 시스템 구성도

2) 양방향 인재매칭 시스템 서비스 프로세스

본 연구에서 제안하는 양방향 인재매칭 시스템 기능 흐름도는 <그림 V-2>와 같이 2가지 주요 기능으로 구성된다.



<그림 V-2> 양방향 인재매칭 시스템 기능 흐름도

첫 번째 주요 기능은 양방향 인재매칭 기능이다. 구직자 사용자와 기업 사용자로 2가지 주요 서비스로 구성된다. 양방향 인재매칭 시스템에 접속하여, 사용자 유형에 맞춰 양방향 추천을 제공한다. 기업 사용자는 해당 기업의 채용정보에 적합한 인재추천을 요청한다. 또한, 구직 희망 사용자는 자신이 작성한 e-포트폴리오에 매칭되는 채용정보 추천을 요청한다. 인재추천 및 채용정보 추천 모두 추천 결과를 Top-N 설정하여 요청할 수 있다. 즉, 자신이 원하는 인재추천 수와 채용정보 추천 수를 설정하여 추천 결과를 전달받을 수 있다.

두 번째 주요 기능은 BERT 기반 양방향 인재매칭 모델의 학습 및 추천 기능이다. 인재매칭 모델의 학습을 위해서 취업 정보 사이트를 주기적으로 크롤링하여 학습 데이터에 반영하는 작업을 진행한다. 이때 크롤링 데이터는 데이터베이스에 저장된다. 데이터베이스에 저장된 데이터를 인재매칭 모델이 학습할 수 있는 데이터 형태로 임베딩 하는 전처리 단계를 거쳐

학습을 주기적으로 진행한다. 인재매칭 추천 기능으로는 인재매칭 시스템에서 양방향 인재매칭 모델 서버로 인재 및 채용정보 Top-N 추천 API 요청을 보내면, 양방향 인재매칭 모델 서버는 요청을 받아 해당 채용정보에 적합한 인재를 추천 또는 구직 희망 사용자의 e-포트폴리오에 적합한 채용정보를 추천하는 양방향 인재매칭 결과를 JSON 형태로 전송한다.

3) 양방향 인재매칭 시스템 API 설계

본 연구에서의 실험 결과를 바탕으로 양방향 인재매칭 모델을 활용한 Web API를 구현하였다. API(Application Programming Interface 애플리케이션 프로그래밍 인터페이스, 응용 프로그램 프로그래밍 인터페이스)는 컴퓨터나 컴퓨터 프로그램 사이의 연결이다. 일종의 소프트웨어 인터페이스이며 다른 종류의 소프트웨어에 서비스를 제공한다. API 설계를 통해 다른 서버와의 연동을 표준화할 수 있도록 기능을 제공한다. API는 파이썬 웹 개발 프레임워크인 FLASK를 기반으로 개발하였다. <표 V-1>은 각 API의 종류와 각 API에 대한 입력과 출력을 설명하고 있다.

<표 V-1> 양방향 인재매칭 Web API

API name	Input Parameters	Output Parameters
/topNHrs (채용정보에 적합한 N명의 인재추천)	num (추천 인재 수) jobID (채용정보 ID)	statusCode (상태코드) msgStr (메시지스트링) recHrSet (인재추천 list)
/topNJobs (구직 희망 사용자에게 적합한 N개의 채용정보 추천)	num (추천 채용정보 수) hrID (구직 희망 사용자 ID)	statusCode (상태코드) msgStr (메시지스트링) recJobSet (채용정보 추천 list)

채용정보에 적합한 N명의 인재추천은 /topNHrs API를 이용할 수 있으며 구직 희망 사용자의 e-포트폴리오에 적합한 N개의 채용정보 추천은 /topNJobs API를 이용할 수 있다. 양방향 인재매칭 모델은 인재매칭 시스템과 데이터베이스 연계되어 있어서 최신의 채용정보 및 희망 구직자의 e-포트폴리오 정보 등 추천에 필요한 데이터 셋을 추가로 업로드 작업을 진행하지 않고 실시간 양방향 인재매칭이 가능하다.

출력 파라미터는 JSON 포맷으로 전송되도록 구현하였다. API를 사용하기 위한 튜토리얼을 정리하였다. <표 V-2>는 채용정보에 적합한 N명의 인재추천을 요청하는 topNHrs API 사용 방법을 정리하였다.

<표 V-2> topNHrs API 사용 방법

1) topNHrs API

- API 호출 방식 예시

<http://hrMatching.kr/topNHrs?num=5&jobID=805>

▷ 입력파라미터 num은 추천받고 싶은 인재의 수

예) num=5이면 요청한 채용정보에 가장 적합한 인재 5명을 추천
이때 5개의 인재는 적합도 순서에 따라 정렬되어 있음

▷ 입력파라미터 jobID는 채용정보 ID

예) jobID=805이면 채용정보 805에 가장 적합한 인재를 추천

▷ topNHrs API 요청 결과 출력은 <표 V-3>와 같다.

num=5인 경우 (JSON 포맷)

▷ JSON 포맷 recHrSet 설명

- {사용자ID: 적합도점수} 로 구성된 Ordered Dictionary 타입
- 적합도 점수 순서로 정렬되어 제공됨
- 최대 N명의 인재가 추천됨

▷ statusCode 정의표는 <표 V-4>과 같다.

<표 V-3> topNHrs API 요청 결과 JSON 포맷

```
{
  "recHrSet": {
    "USER001": 0.9985185282933657,
    "USER005": 0.9582677819881763,
    "USER008": 0.9480728820535575,
    "USER010": 0.9378542805036725,
    "USER022": 0.9375045967902668
  },
  "msgStr": "successfully recommended!",
  "statusCode": "100"
}
```

<표 V-4> topNHrs API statusCode 정의표

API name	Status Code	Comment
topNHrs (채용정보에 적합한 N명의 인재추천)	100	정상 종료
	101	입력값 오류
	102	인재추천 불가

<표 V-5>는 구직 희망 사용자에게 적합한 N개의 채용정보 추천을 요청하는 topNJobs API 사용 방법을 정리하였다.

<표 V-5> topNJobs API 사용 방법

2) topNJobs API

- API 호출 방식 예시

`http://hrMatching.kr/topNJobs?num=5&hrID=user08`

▷ 입력파라미터 num은 추천받고 싶은 채용정보의 수

예) num=5이면 요청한 구직 희망 사용자에게 가장 적합한 채용정보 5건을 추천

이때 5건의 채용정보는 적합도 순서에 따라 정렬되어 있음

▷ 입력파라미터 hrID는 구직 희망 사용자 ID

예) hrID=user08이면 구직 희망 사용자 user08에게 가장 적합한 채용정보를 추천

▷ topNJobs API 요청 결과 출력은 <표 V-6>과 같다.

num=5인 경우 (JSON 포맷)

▷ JSON 포맷 recJobSet 설명

- {채용정보ID: 적합도점수}로 구성된 Ordered Dictionary 타입
- 적합도 점수 순서로 정렬되어 제공됨
- 최대 N 건의 채용정보가 추천됨

▷ statusCode 정의표는 <표 V-7>과 같다.

<표 V-6> topNJobs API 요청 결과 JSON 포맷

```
{
  "recJobSet": {
    "55": 0.9985185282933657,
    "80": 0.9582677819881763,
    "30": 0.9480728820535575,
    "28": 0.9378542805036725,
    "8": 0.9375045967902668
  },
  "msgStr": "successfully recommended!",
  "statusCode": "100"
}
```

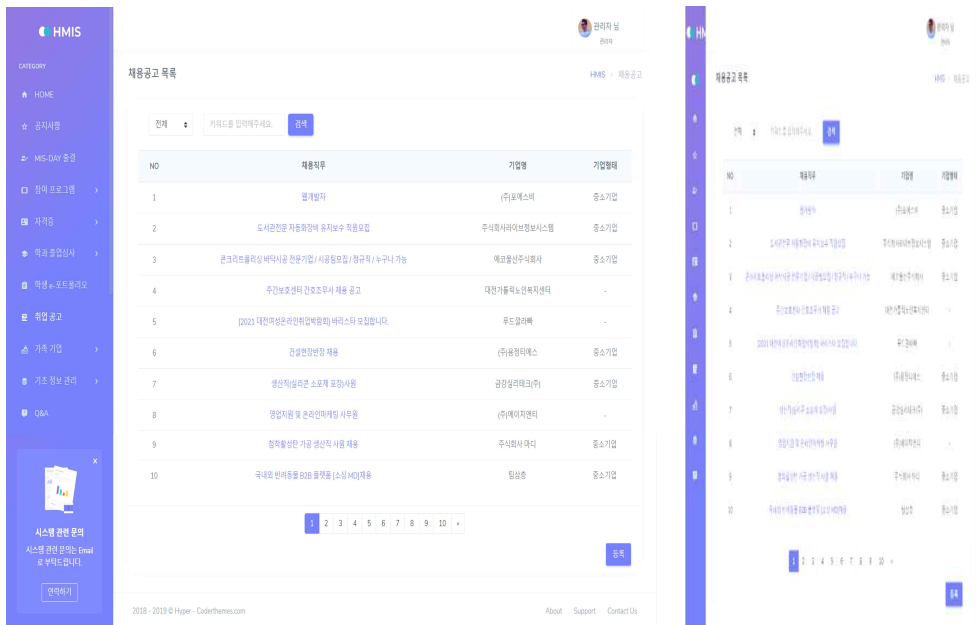
<표 V-7> topNJobs API statusCode 정의표

API name	Status Code	Comment
topNJobs (구직 희망 사용자에게 적합한 N 건의 채용정보 추천)	100	정상 종료
	101	입력값 오류
	102	채용정보 추천 불가

4) 양방향 인재매칭 시스템 예상 결과물

본 연구에서 제안하는 양방향 인재매칭 시스템 예상 결과물은 인재매칭 화면에서 제안 모델의 Top-N 추천의 성능평가 Top-N 추천 재현율 (REC@N)과 Top-N 추천 정확도(AAS@N)를 활용하여 Top N 기본값을 25로 설정한다. 양방향 인재매칭 시스템 결과물은 기업 채용정보 목록 화면, 기업 사용자 인재매칭 화면, 구직자 e-포트폴리오 화면, 구직자 채용정보 매칭 화면 총 4가지 기능으로 화면 설계하였다.

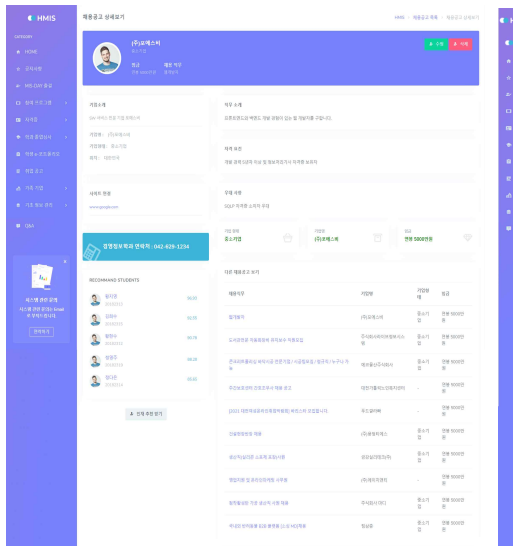
UI(User Interface)는 반응형 웹 서비스로 개발하였다. 반응형 웹서비스는 디바이스 크기에 최적화된 UI를 제공하는 기술이다. UI를 의도대로 보여주면서 동시에 디스플레이 되는 기기의 요구사항(해상도, 브라우저 등)에 유연하게 반응할 수 있도록 하여, 반응형 웹 서비스로 모든 디바이스에서 편리하게 사용하도록 설계하였다. <그림 V-3> ~ <그림 V-6>은 4가지 기능으로 화면 설계한 예상 결과물을 (a) 웹 브라우저 접속 화면과 (b) 모바일 접속 화면으로 나타내었다.



(a) 웹 브라우저 접속 화면

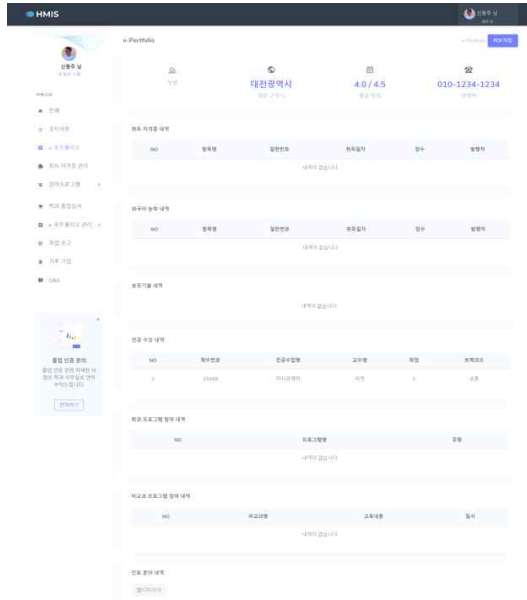
(b) 모바일 접속 화면

<그림 V-3> 기업 채용정보 목록 UI




(a) 웹 브라우저 접속 화면 (b) 모바일 접속 화면

<그림 V-4> 기업 사용자 인재매칭 UI



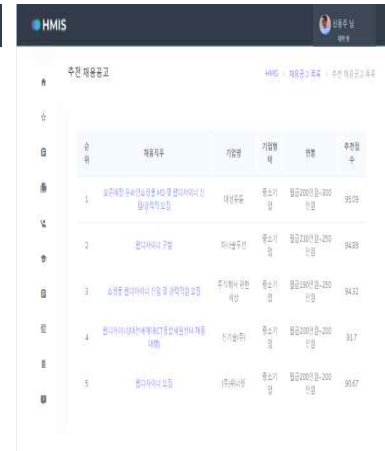
(a) 웹 브라우저 접속 화면 (b) 모바일 접속 화면

<그림 V-5> 구직자 e-포트폴리오 UI



주최 채용공고

순위	채용직무	기업명	기업형태	연봉	추천점수
1	요양병원 온라인시스템을 MBS를 활용하여서 신입강의자 모집	대성병원	종소기업	월급200만원-300만원	95.09
2	웹디자인의 구형	해나솔루션	종소기업	월급230만원-250만원	94.89
3	소형물품디자인의 신입 및 경력직의 모집	주식회사 편편세상	종소기업	월급180만원-250만원	94.32
4	웹디자인에서전통적인디자인을 대체할 새로운 디자인	신기솔루션	종소기업	월급200만원-300만원	91.7
5	웹디자인의 모집	(주)유니넷	종소기업	월급200만원-300만원	90.67



주최 채용공고

순위	채용직무	기업명	기업형태	연봉	추천점수
1	요양병원 온라인시스템을 MBS를 활용하여서 신입강의자 모집	대성병원	종소기업	월급200만원-300만원	95.09
2	웹디자인의 구형	해나솔루션	종소기업	월급230만원-250만원	94.89
3	소형물품디자인의 신입 및 경력직의 모집	주식회사 편편세상	종소기업	월급180만원-250만원	94.32
4	웹디자인에서전통적인디자인을 대체할 새로운 디자인	신기솔루션	종소기업	월급200만원-300만원	91.7
5	웹디자인의 모집	(주)유니넷	종소기업	월급200만원-300만원	90.67

(a) 웹 브라우저 접속 화면 (b) 모바일 접속 화면

<그림 V-6> 구직자 채용정보 매칭 UI

기업 사용자는 <그림 V-3>과 같이 기업의 채용정보를 등록하고, 다른 기업의 채용정보도 열람할 수 있다. 또한 <그림 V-4>와 같이 기업이 등록한 채용정보에 적합한 인재를 채용정보 상세보기 화면에서 추천받을 수 있다.

구직자는 <그림 V-5>와 같이 자신의 이력서와 경력 사항들을 e-포트폴리오에 등록하고, e-포트폴리오에 작성된 자신이 보유한 기술 및 직무 능력에 적합한 채용정보를 <그림 V-6>과 같이 추천받을 수 있다. 사용자 화면은 반응형 웹 서비스로 구축하여, 웹 서비스뿐만 아니라 안드로이드, IOS에 맞는 하이브리드 앱으로도 발전할 수 있다.

VI. 결 론

1. 연구의 요약 및 공헌

본 연구에서는 특정 기업의 채용정보에 대해 요구되는 역량과 기술을 보유한 구직자를 추천하는 것과 구직자가 보유한 역량과 관심에 적합한 채용정보를 추천하는 양방향 인재매칭을 위해 BERT 기반의 전이학습 모델을 이용한 양방향 인재매칭 모델을 제안하였다. 제안 모델은 기존 내용 기반 척도학습모형보다 정밀도와 재현율, f1-score, AUC, Top-N 추천인 성능평가 Top-N 추천 재현율(REC@N)과 Top-N 추천 정확도(AAS@N)에서 높은 성과를 나타내었다. 특히, 재현율을 볼 때 학습되지 않는 새로운 데이터인 구직자의 신규 이력서 및 채용정보의 의미를 분석하여 추천하는 부분에서 전이학습 모델이 높은 성능을 낼 수 있다는 것을 보여주고 있다.

본 연구는 다양한 측면에서 의의를 가진다. BERT 모델을 이용한 양방향 인재매칭 모델을 제안하였으며, 제안 모델은 양방향 추천이 가능할 뿐 아니라, 전이학습 모델을 사용하여 문맥적인 의미와 관계가 무시되었던 문제점을 해결한다. 기존 연구에서 활용되었던 bag-of-words, TF-IDF 등과 같은 자체적인 사전 구축을 통한 워드 임베딩은 최소 행렬로 학습 시 계산의 복잡성이 증가하고 단어의 순서를 고려하지 않기 때문에 문맥적인 의미를 파악하기 힘든 문제가 있다. 사전에 구축되지 않은 단어 줄임말이나, 의미는 같으나 다른 단어로 되어 있는 경우 의미적 관계 파악하기 어려웠던 문제점 또한 제안 모델에 전이학습 모델인 BERT를 적용

하여 해결할 수 있었다. 구직자의 이력서의 전체 텍스트와 채용정보 등과 같은 긴 장문의 텍스트 데이터 전처리 작업을 BERT에서 제공하는 임베딩을 통해 별다른 전처리 작업을 거치지 않고 토큰(Token), 인덱스(Index), 세그먼트와 포지션(Segment & Position)으로 쉽게 데이터를 학습시킬 수 있었다. 그리고 추천 시스템에서 가장 큰 문제 중 하나인 과거 이력이 존재하지 않는 새로운 사용자나 아이템에 대해서는 추천할 수 없는 콜드 스타트 문제 (cold-start problem)도 BERT 모델을 통해 해결할 수 있었다.

BERT 모델은 문장을 기반으로 언어 표현을 이해하기 때문에 신규 구직자의 이력서와 기업의 신규 채용정보에 대해서 재학습 없이 양방향 인재매칭이 가능하기 때문이다. 제안 모델은 자체적인 용어 사전 구축을 위한 많은 학습 데이터의 획득을 위해 비용이 많이 들어가는 인재매칭 분야에서 전이학습 모델을 적용하여 데이터 획득 비용을 줄이고 매칭 성능을 향상시킬 수 있는 효과적인 방법으로 평가되었다.

BERT를 적용한 추천 모델의 설계와 성능평가를 통해 활발한 연구가 이루어지고 있지만 제안된 모델을 실제 시스템으로 구축하거나, 시스템 구성을 제안하는 사례는 드물다. BERT를 적용한 모델의 경우 기존 웹 시스템 구축에서 사용되는 일반적인 프로그래밍 언어(java, php, asp 등)와 다른 프로그래밍 언어인 파이썬(Python) 또는 파이토치(PyTorch)를 사용하여 개발되기 때문에 웹 서비스를 위한 시스템과 추천 모델과 연동하기 위한 시스템 설계는 필수적이다. 본 연구에서는 서로 다른 프로그래밍 언어로 되어 있는 추천 모델과 웹 서비스 시스템을 연동할 수 있는 시스템 설계와 구성도를 제안하였으며, 다른 시스템에서도 추천 모델을 연동

할 수 있도록 공통의 API를 설계하였다. 이를 통해 새로 시스템을 구축
가능할 뿐만 아니라, 기존의 구인 구직 사이트에서도 API를 통해 본 연
구에서 제안한 추천 모델과 연계하여 양방향 인재추천이 가능하다.

2. 연구의 한계 및 향후 연구

본 연구에서 제안한 양방향 인재매칭 시스템 구성도를 반영한 실제 시스템을 구축하여 유용성 확인을 하지 못한 한계점을 갖는다. 추천 결과에 따른 사용자로부터 만족도 조사를 통해 추천 결과 유용성을 확인할 수 있도록 화면 설계 및 데이터베이스 확장이 필요하다. 또한, 인재매칭 및 일자리매칭 결과에 따른 면접/채용에 대한 피드백을 추후 BERT 기반의 전이학습 모델에 재학습 시킬 수 있도록 데이터베이스 확장이 필요하다.

향후 연구로 실제 양방향 인재매칭 시스템을 구축하고 운영하면서 기업과 구직자에게 추천 결과에 대한 만족도 조사와 실제 면접 및 채용으로 연계되었는지에 대한 피드백을 통해 시스템의 유용성 확인을 진행할 것이다.

또한, 본 연구의 핵심인 양방향 인재매칭 모델의 성능 개선을 위해 하이퍼 파라미터 튜닝이 가능할 것이다. 제안 모델은 4회의 훈련을 통해 높은 추천 성능을 보여주었으나 여전히 개선 여지가 남아 있고 매칭 오류를 개선하기 위해서는 옵티마이저(Optimizer), 학습률(Learning Rate), 학습률 스케줄링 방법, 손실함수(Loss Function), 훈련 반복횟수 훈련(Epoch)과 배치사이즈(Batch size)의 변경과 같은 경험적 실험을 통해 최적의 설정을 위한 하이퍼 파라미터 튜닝이 가능할 것이다. 그리고 한국어 인재매칭에 더욱 특화된 모델 개발을 위해 BERT 기본적으로 제공하는 ‘BERT-Base, Multilingual Cased’ 모델 외에 대량의 한국어 코퍼스를 통해 기존 BERT 다국어 모델을 재학습시킨 한국어 언어 모델(Korean Pre-trained Language Models)인 SKTBrain의 KoBERT 모델, ETRI의 KorBERT 모델을 적용하여 성능을 비교하고, 지속적인 성능

개선을 진행할 수 있을 것이다. 또한, 채용정보와 인재매칭 시 조건을 활용한 필터링 기능 등 정량적 구간 조건 비교 기능이 추천 시스템과 결합된다면 정교한 추천 및 매칭 정확도를 높일 수 있을 것이다.

본 연구에서 제안한 BERT 기반의 전이학습 모델을 적용한 양방향 인재매칭 모델이 인재매칭 시스템이 상용화된다면 구직자와 기업 모두에게 개인화된 일자리와 인재를 양방향으로 추천하는 서비스를 제공할 수 있을 것이다. 특히 ‘체감 실업률’이 사상 최악 수준을 경험하고 있는 코로나 19세대의 청년들의 취업난과 유례없는 구인난을 겪고 있는 중소기업의 일자리 미스매치를 해소하고 국가의 고질적인 취업난을 해결하는데 이바지할 수 있을 것으로 기대한다.

참 고 문 헌

[국내문헌]

김지운. (2018). 2014 년 이후 실업률 상승에 대한 요인 분석. [KDI] 경제전망, 35(2).

박호연, & 김경재. (2021). BERT 기반 감성분석을 이용한 추천시스템. 지능정보연구, 27(2), 1-15.

손배원. (2015). NCS 기반 채용시스템 사례 연구: 한국산업인력공단의 NCS 기반 채용을 중심으로. 경영컨설팅연구, 15(4), 217-228.

송지수. (2020). BERT 를 이용한 불안 분류기 기반의 사회적 불안 시각화 시스템 및 실제 사례 해석 (Doctoral dissertation, 한양대학교).

송희석. (2020). 인재매칭을 위한 내용기반 척도학습모형의 설계. Journal of Information Technology Applications & Management, 27(6), 141-151.

유소엽, & 정옥란. (2019). BERT 모델과 지식 그래프를 활용한 지능형 챗봇. 한국전자거래학회지, 24(3), 87-98.

유승의.(2021) 인공지능과 자연어 처리 기술 동향. ITFIND 주간기술동향 1984호.

이용주, 문용혁, 박준용, 민옥기. (2019). 경량 딥러닝 기술 동향.
Electronics and Telecommunications Trends, 34(2), 40-50.
<https://doi.org/10.22648/ETRI.2019.J.340205>

한억수 & 심진보 (2019). AI-based Job Mismatch Response
Service Trends and Strategic Challenges. 한국콘텐츠학회
ICCC 논문집, 159-160.

한재현, 김재희, 전원호, 김형주, & 홍윤식. (2020). BERT 언어 모델을
사용한 상담 매칭 시스템. 한국정보과학회 학술발표논문집,
1566-1568.

홍승연, 나승훈, 김고운, 신병수, & 정영철. (2020). BERT 를 이용한 알
츠하이머병 치매와 조현병 진단. 한국정보과학회 학술발표논문집,
419-421.

통계청 (2021) <http://kostat.go.kr/>

한국경영자총협회 (2021), “최근 고용 흐름의 3가지 특징과 시사점
(2021년 3분기)”

행정안전부 (2021), “기재부 등 4개 주요부처에 ‘청년’ 전담부서 생긴
다” 보도자료, 2021.08.31.

ETRI (2019), https://aiopen.etri.re.kr/service_dataset.php

SKTBrain (2020), <https://github.com/SKTBrain/KoBERT>

[해외문헌]

- Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., & Marchand, M. (2014). Domain-adversarial neural networks. arXiv preprint arXiv:1412.4446.
- Arjovsky, M., Chintala, S., & Bottou, L. (2017, July). Wasserstein generative adversarial networks. In International conference on machine learning (pp. 214–223). PMLR.
- Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.
- Baralis, E., Chiusano, S., & Garza, P. (2007). A lazy approach to associative classification. *IEEE Transactions on Knowledge and Data Engineering*, 20(2), 156–171.
- Bigdeli, E., & Bahmani, Z. (2008, September). Comparing accuracy of cosine-based similarity and correlation-based similarity algorithms in tourism recommender systems. In 2008 4th IEEE International Conference on Management of Innovation and Technology (pp. 469–474). IEEE.
- Blum, A., & Mitchell, T. (1998, July). Combining labeled and unlabeled data with co-training. In Proceedings of the eleventh annual conference on Computational learning

theory (pp. 92–100).

Bobadilla, J., Ortega, F., Hernando, A., & Bernal, J. (2012). A collaborative filtering approach to mitigate the new user cold start problem. *Knowledge-based systems*, 26, 225–238.

Bobadilla, J., Ortega, F., Hernando, A., and Guti rrez, A. (2013), Recommender system s survey, *Knowledge-Based Systems*, 46, 109–132.

Cambria, E., & White, B. (2014). Jumping NLP curves: A review of natural language processing research. *IEEE Computational intelligence magazine*, 9(2), 48–57.

Chang, H., Han, J., Zhong, C., Snijders, A. M., & Mao, J. H. (2017). Unsupervised transfer learning via multi-scale convolutional sparse coding for biomedical applications. *IEEE transactions on pattern analysis and machine intelligence*, 40(5), 1182–1194.

Cheng, J., Dong, L., & Lapata, M. (2016). Long short-term memory-networks for machine reading. *arXiv preprint arXiv:1601.06733*.

Cho, K., Van Merri nboer, B., Bahdanau, D., & Bengio, Y. (2014). On the properties of neural machine translation:

Encoder-decoder approaches. arXiv preprint arXiv:1409.1259.

Choi, D., Shallue, C. J., Nado, Z., Lee, J., Maddison, C. J., & Dahl, G. E. (2019). On empirical comparisons of optimizers for deep learning. arXiv preprint arXiv:1910.05446.

Dacrema, M. F., Cremonesi, P., & Jannach, D. (2019, September). Are we really making much progress? A worrying analysis of recent neural recommendation approaches. In Proceedings of the 13th ACM Conference on Recommender Systems (pp. 101–109).

Dai, W., Yang, Q., Xue, G.R., Yu, Y. (2007). Boosting for transfer learning. In Proceedings of the 24th international conference on Machine learning (pp. 193–200). ACM.

Das, M., De Francisci Morales, G., Gionis, A., and Weber, I. (2013), Learning to question : Leveraging user preferences for shopping advice, In Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining, 203–211.

De Pessemier, T., Vanhecke, K., & Martens, L. (2016). A scalable, high-performance Algorithm for hybrid job recommendations. In Proceedings of the Recommender Systems Challenge (pp. 1–4).

- Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.
- Ganin, Y., & Lempitsky, V. (2015, June). Unsupervised domain adaptation by backpropagation. In International conference on machine learning (pp. 1180–1189). PMLR.
- George, D., Shen, H., & Huerta, E. A. (2017). Deep Transfer Learning: A new deep learning glitch classification method for advanced LIGO. arXiv preprint arXiv:1706.07446.
- Goldberg, D., Nichols, D., Oki, B. M., & Terry, D. (1992). Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, 35(12), 61–70.
- Gretton, A., Sejdinovic, D., Strathmann, H., Balakrishnan, S., Pontil, M., Fukumizu, K., & Sriperumbudur, B. K. (2012). Optimal kernel choice for large-scale two-sample tests. In *Advances in neural information processing systems* (pp. 1205–1213).
- Hochreiter, S., Bengio, Y., Frasconi, P., & Schmidhuber, J. (2001). Gradient flow in recurrent nets: the difficulty of learning long-term dependencies.
- Huang, J. T., Li, J., Yu, D., Deng, L., & Gong, Y. (2013, May).

- Cross-language knowledge transfer using multilingual deep neural network with shared hidden layers. In 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (pp. 7304–7308). IEEE.
- Joachims, T. (1999, June). Transductive inference for text classification using support vector machines. In *Icml* (Vol. 99, pp. 200–209).
- Kenthapadi, K., Le, B., & Venkataraman, G. (2017, August). Personalized job recommendation system at linkedin: Practical challenges and lessons learned. In *Proceedings of the eleventh ACM conference on recommender systems* (pp. 346–347).
- Kim, Y., Denton, C., Hoang, L., & Rush, A. M. (2017). Structured attention networks. *arXiv preprint arXiv:1702.00887*.
- Konstan, J. A., Miller, B. N., Maltz, D., Herlocker, J. L., Gordon, L. R., and Riedl, J. (1997), GroupLens : applying collaborative filtering to Usenet news, *Communications of the ACM*, 40(3), 77–87.
- Kuncheva, L. I., & Rodriguez, J. J. (2007). Classifier ensembles with a random linear oracle. *IEEE Transactions on Knowledge and Data Engineering*, 19(4), 500–508.

- Lang, K. (1995), Newsweeder : Learning to filter netnews, In Proceedings of the Twelfth International Conference on Machine Learning.
- Le, Q., & Mikolov, T. (2014, June). Distributed representations of sentences and documents. In International conference on machine learning (pp. 1188–1196). PMLR.
- Leksins, V., & Ostapets, A. (2016). Job recommendation based on factorization machine and topic modelling. In Proceedings of the Recommender Systems Challenge (pp. 1–4).
- Li, N., Hao, H., Gu, Q., Wang, D., & Hu, X. (2017). A transfer learning method for automatic identification of sandstone microscopic images. Computers & Geosciences, 103, 111–121.
- Lin, Z., Feng, M., Santos, C. N. D., Yu, M., Xiang, B., Zhou, B., & Bengio, Y. (2017). A structured self-attentive sentence embedding. arXiv preprint arXiv:1703.03130.
- Liu, X., Liu, Z., Wang, G., Cai, Z., & Zhang, H. (2017). Ensemble transfer learning algorithm. IEEE Access, 6, 2389–2396.
- Long, M., Cao, Y., Wang, J., & Jordan, M. (2015, June). Learning transferable features with deep adaptation networks. In International conference on machine learning (pp. 97–105).

PMLR.

Long, M., Cao, Z., Wang, J., & Jordan, M. I. (2017). Domain adaptation with randomized multilinear adversarial networks. arXiv preprint arXiv:1705.10667.

Long, M., Zhu, H., Wang, J., & Jordan, M. I. (2016). Unsupervised domain adaptation with residual transfer networks. arXiv preprint arXiv:1602.04433.

Long, M., Zhu, H., Wang, J., & Jordan, M. I. (2017, July). Deep transfer learning with joint adaptation networks. In International conference on machine learning (pp. 2208–2217). PMLR.

Luo, Z., Zou, Y., Hoffman, J., & Fei-Fei, L. (2017). Label efficient learning of transferable representations across domains and tasks. arXiv preprint arXiv:1712.00123.

Luong, M. T., Pham, H., & Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. arXiv preprint arXiv:1508.04025.

Maheshwary, S., & Misra, H. (2018, April). Matching resumes to jobs via deep siamese network. In Companion Proceedings of the The Web Conference 2018 (pp. 87–88).

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient

estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

Mikolov, T., Karafiat, M., Burget, L., Cernocký, J., & Khudanpur, S. (2010, September). Recurrent neural network based language model. In Interspeech (Vol. 2, No. 3, pp. 1045–1048).

Nigam, K., McCallum, A. K., Thrun, S., & Mitchell, T. (2000). Text classification from labeled and unlabeled documents using EM. *Machine learning*, 39(2), 103–134.

Oquab, M., Bottou, L., Laptev, I., & Sivic, J. (2014). Learning and transferring mid-level image representations using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1717–1724).

Pan, S. J., & Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10), 1345–1359.

Pan, S. J., & Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10), 1345–1359.

Pardoe, D., & Stone, P. (2010, January). Boosting for regression

transfer. In Proceedings of the 27th International Conference on International Conference on Machine Learning (pp. 863-870).

Parikh, A. P., Töckström, O., Das, D., & Uszkoreit, J. (2016). A decomposable attention model for natural language inference. arXiv preprint arXiv:1606.01933.

Paulus, R., Xiong, C., & Socher, R. (2017). A deep reinforced model for abstractive summarization. arXiv preprint arXiv:1705.04304.

Pazzani, M. J. and Billsus, D. (2007), Content-based recommendation systems, In The adaptive web, 325-341.

Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., and Riedl, J. (1994), GroupLens : an open architecture for collaborative filtering of netnews, In Proceedings of the ACM conference on Computer supported cooperative work, 175-186.

Ruder, S. (2016). An overview of gradient descent optimization algorithms. arXiv preprint arXiv:1609.04747.

Schafer, J. B., Konstan, J. A., and Riedl, J. (2001), E-commerce recommendation applications, In Applications of Data Mining to Electronic Commerce, 115-153.

- Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A. Y., & Potts, C. (2013, October). Recursive deep models for semantic compositionality over a sentiment treebank. In Proceedings of the 2013 conference on empirical methods in natural language processing (pp. 1631–1642).
- Sohrabi, B., Vanani, I. R., & Shineh, M. B. (2018). Topic modeling and classification of cyberspace papers using text mining. *Journal of Cyberspace Studies*, 2(1), 103–125.
- Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In Advances in neural information processing systems (pp. 3104–3112).
- Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., & Liu, C. (2018, October). A survey on deep transfer learning. In International conference on artificial neural networks (pp. 270–279). Springer, Cham.
- Tzeng, E., Hoffman, J., Darrell, T., & Saenko, K. (2015). Simultaneous deep transfer across domains and tasks. In Proceedings of the IEEE international conference on computer vision (pp. 4068–4076).
- Tzeng, E., Hoffman, J., Saenko, K., & Darrell, T. (2017). Adversarial discriminative domain adaptation. In Proceedings of the IEEE conference on computer vision and pattern

recognition (pp. 7167–7176).

Tzeng, E., Hoffman, J., Zhang, N., Saenko, K., & Darrell, T. (2014). Deep domain confusion: Maximizing for domain invariance. arXiv preprint arXiv:1412.3474.

Valverde–Rebaza, J. C., Puma, R., Bustios, P., & Silva, N. C. (2018). Job Recommendation Based on Job Seeker Skills: An Empirical Study. In Text2Story@ ECIR (pp. 47–51).

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In Advances in neural information processing systems (pp. 5998–6008).

Wan, C., Pan, R., & Li, J. (2011, June). Bi-weighting domain adaptation for cross-language text classification. In Twenty-Second International Joint Conference on Artificial Intelligence (pp. 1535).

Weiss, K., Khoshgoftaar, T. M., & Wang, D. (2016). A survey of transfer learning. Journal of Big data, 3(1), 1–40.

Wu, Y. H. and Chen, A. L. (2000), Index structures of user profiles for efficient web page filtering services, In 2012 IEEE 32nd International Conference on Distributed Computing Systems, 644–644.

- Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., ... & Dean, J. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. arXiv preprint arXiv:1609.08144.
- X. Zhu.(2006) Semi-supervised learning literature survey, University of Wisconsin-Madison, Tech. Rep. 1530.
- Xu, Y., Pan, S. J., Xiong, H., Wu, Q., Luo, R., Min, H., & Song, H. (2017). A unified framework for metric transfer learning. IEEE Transactions on Knowledge and Data Engineering, 29(6), 1158–1171.
- Yang, Q., Ling, C., Chai, X., & Pan, R. (2006). Test-cost sensitive classification on data with missing values. IEEE Transactions on Knowledge and Data Engineering, 18(5), 626–638.
- Yao, Y., & Doretto, G. (2010, June). Boosting for transfer learning with multiple sources. In 2010 IEEE computer society conference on computer vision and pattern recognition (pp. 1855–1862). IEEE.
- Yin, X., Han, J., Yang, J., & Yu, P. S. (2006). Efficient classification across multiple database relations: A crossmine approach. IEEE Transactions on Knowledge and Data Engineering, 18(6), 770–783.

- Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). How transferable are features in deep neural networks?. arXiv preprint arXiv:1411.1792.
- Young, T., Hazarika, D., Poria, S., & Cambria, E. (2018). Recent trends in deep learning based natural language processing. *IEEE Computational Intelligence Magazine*, 13(3), 55–75.
- Zhu, H., Long, M., Wang, J., & Cao, Y. (2016, March). Deep hashing network for efficient similarity retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 30, No. 1).
- Zhu, X., & Wu, X. (2006). Class noise handling for effective cost-sensitive learning by cost-guided iterative classification filtering. *IEEE Transactions on Knowledge and Data Engineering*, 18(10), 1435–1440.
- Bobadilla, J., Ortega, F., Hernando, A., & Bernal, J. (2012). A collaborative filtering approach to mitigate the new user cold start problem. *Knowledge-based systems*, 26, 225–238.

ABSTRACT

Bidirectional HR Matching system Using a BERT-based Transfer Learning Model

Oh, So Jin

Department of Management Information Systems
Graduate School of Hannam University
(Supervised by professor Song, Hee Seok)

The most fundamental cause of the labor shortage in companies and youth unemployment is the job mismatch phenomenon in which there is a discrepancy between the type of job competency that companies expect job applicants to have and the job competency job seekers possess. In order to resolve the job mismatch, an bidirectional matching system is required to recommend a job vacancy based on the job seeker's capabilities and interests and vice versa.

In this study, BERT(Bidirectional Encoder Representations from Transformers)-based job matching algorithm is developed to deal with the job mismatch. BERT is a transfer learning model that has recently shown excellent performance in the field of machine

language translation and natural language processing and the proposed BERT-based transfer learning model provides bidirectional recommendations for the company and job applicants based on the contents of job posting and the job applicant's resume. An information system design is also proposed to implement of for the proposed bidirectional matching model to the actual system.

By applying the BERT-based Multilingual Cased' model that supports 104 different languages, the proposed algorithm tokenizes contents of job posting and the job applicant's profile (resume) and use them as inputs of BERT pre-trained language model and a fine-tuning model produces the indices of goodness-of-fit between job position and job applicants as the final output layer.

Hyperparameter setting of the model is as follows. Rectified Adam (RAdam) is the optimizer, the learning rate is $1.0e-5$, and binary cross entropy is the loss function since matching is a binary classification problem (fit/nonconformity). A weight decay is set to 0.0025 to solve the overfitting problem, and the epoch (number of iterations of model training) is set to 4. Python (version 3.7) and Google Colab's TPU were used for model implementation.

The results of the proposed algorithm for precision, recall and

f1-score was higher than those of the existing content-based metric learning model for matching job positions and job applicants.

The system design proposed in this study includes common application programming interfaces(API) that link the BERT-based job matching model with existing job matching services developed in various programming languages. The /topNHrs API is to recommend N applicants matched for a job position, and the /topNJobs API is to recommend N job positions suitable for a job applicant.

Bidirectional HR matching model proposed in this study could lower the search cost of both companies and job applicants as companies can reduce additional cost such as hiring headhunters and advertisements and job applicants also reduce time and effort in applying less relevant jobs to their competencies and interests, resulting in resolving the job mismatch.

Keywords : HR Matching, HR Matching System, Job Recommendation, BERT, Transfer Learning, Pre-trained Language Model, Fine-tuning Deep Learning Model, Natural Language Processing