뉴스 클러스터링을 위한 문장 간 상호 작용 기반 문서 쌍 유사도 측정 모델들

최성환[O1], 손동현[1], 이호창[2]

서울대학교 컴퓨터공학부[1], 엔씨소프트[2]
csh3695@snu.ac.kr[1], happydh1@snu.ac.kr[1], hochang@ncsoft.com[2]

Sentence Interaction-based Document Similarity Models for News Clustering

Seonghwan Choi[O1], Donghyun Son[1], Hochang Lee[2]
Dept. of Computer Science and Engineering, Seoul National University[1], NCSOFT[2]

요 약

뉴스 클러스터링에서 두 문서 간의 유사도는 클러스터의 특성을 결정하는 중요한 부분 중 하나이다. 전통적인 단어 기반 접근 방법인 TF-IDF 벡터 유사도는 문서 간의 의미적인 유사도를 반영하지 못하고, 기존 딥러닝 기반 접근 방법인 시퀀스 유사도 측정 모델은 문서 단위에서 나타나는 긴 문맥을 반영하지 못하는 문제점을 가지고 있다. 이 논문에서 우리는 뉴스 클러스터링에 적합한 문서 쌍 유사도 모델을 구성하기 위하여 문서 쌍에서 생성되는 다수의 문장 표현들 간의 유사도 정보를 종합하여 전체 문서 쌍의 유사도를 측정하는 네 가지 유사도 모델을 제안하였다. 이 접근 방법들은 하나의 벡터로 전체 문서 표현을 압축하는 HAN (hierarchical attention network)와 같은 접근 방법에 비해 두 문서에서 나타나는 문장들 간의 직접적인 유사도를 통해서 전체 문서 쌍의 유사도를 추정한다. 그리고 기존 접근 방법들인 SVM과 HAN과 제안하는 네 가지 유사도 모델을 통해서 두 문서 쌍 간의 유사도 측정 실험을 하였고, 두 가지 접근 방법에서 기존 접근 방법들보다 높은 성능이 나타나는 것을 확인할 수 있었고, 그래프 기반 접근 방법과 유사한 성능을 보이지만 더 효율적으로 문서 유사도를 측정하는 것을 확인하였다.

주제어: 뉴스 클러스터링, 문서 쌍 유사도, 텍스트 유사도

1. 서론

문서 클러스터링 [1]은 전통적인 텍스트 처리 태스크 중 하나로, 일반적으로 비지도 학습 접근 방법 문제이다. 하지만 동일한 사건 단위로 뉴스들을 그룹화하는 뉴스 클러스터링과 같이 원하는 형태의 클러스터 결과가 있을 때, 준지도학습 기반 클러스터링 [2]은 선택할 수 있는 자연스러운 옵션 중 하나이다. 이 준지도학습 기반 클러스터링 중 하나인 거리 함수 (distance-function) 학습 접근 방법은 클러스터링 여부를 결정하는데 중요한 거리 함수 또는 유사도 함수를 수작업으로 태깅된 데이터 집합을 통해서 구축하고, 이 거리 함수를 기반으로 클러스터링의 결과를 원하는 방향으로 유도한다. 따라서 이와 같은 접근 방법에서 지도학습으로 학습되는 거리 함수가 전체 클러스터링의 성능을 결정하는 중요한 요소가 된다.

문서 간의 관계 또는 유사도를 판별하는 것은 중요한 자연 언어 처리 기술 중 하나이고, 대량의 문서 집합을 처리하는 뉴스 시스템과 정보 검색 기술 등의 분야에서 중복을 제거하거나 유사한 문서 별로 클러스터링 하기 위해 사용된다. 하지만 문장에 비해 일반적으로 긴 문서 단위의 관계는 각 문서가 다양한 토픽을 포함할 수 있고, 복잡한 문단 구조와 풍부한 의미 정보를 가질 수 있기 때문에 기존의 문장 또는 짧은 텍스트 단위의 접근 방법보다 어려운 태스크이다.

전통적인 단어 벡터 기반의 접근 방법인 TF-IDF나 BM25 [3]같은 방법들은 입력 텍스트의 각 단어의 매칭 비율을 기준으로 텍스트 간의 유사도를 측정하는 반면, 최근 딥러닝 기반 접근 방법들 [4-13]은 RNN이나 CNN 같은 고차원의 딥 뉴럴 네트워크를 통해서 텍스트 내의 의미적인 유사도를 추정하고자 하였다. 텍스트 간의 의미적인 유사도를 측정하는 딥러닝 기반의 연구들은 크게 두 가지로 구분할 수 있다: 텍스트 표현 기반 접근 방법, 상호 작용 (Interaction) 기반 접근 방법. 텍스트 표현 기반 접근 방법은 CNN [4], RNN, HAN [5]와 같은 딥 뉴럴 네트워크를 통해서 입력된 문서를 하나의 벡터 표현으로 변환하고, 상호 작용 기반 접근 방법 [6,9-13]은 하나의 벡터를 구성하지 않고, 하위 텍스트 단위 (예를 들어, 문장 또는 단어) 간의 복잡한 상호 관계를 기반으로 전체 문서의 유사성을 추론한다. 텍스트 표현 기반 접근 방법은 단일 문서 표현 벡터가 전체 문서의 복잡한 문맥 정보를 잘 표현하지 못하는 문제점을 가지고 있고, 상호 작용 기반 접근 방법은 하위 텍스트 단위의 표현을 사용하여 이러한 문제점을 보완하고자 하였지만 대부분의 연구가 문장 단위 유사도 측정에 머물러 있다.

본 연구에서 뉴스 클러스터링에 적합한 문서 쌍 유사도 측정 모델을 구축하기 위하여 문서 쌍에서 나타나는 다수의 문장 표현들 간의 유사도 정보를 종합하여 전체 문서 쌍의 유사도를 추정하는 네 가지 유사도 모델을 제

안한다. 이 모델들은 기존의 상호 작용 기반 접근 방법들 [6-13]과 달리 단어-문장-문서의 구조를 가지고 있는 문서 단위의 유사도를 측정하기 위하여 문장 벡터 간의 상호 작용들을 통해서 문서 간의 유사도를 측정한다. 먼저 이 모델에서 우리는 HAN과 동일하게 어텐션 기반 단어 인코더로 문장 벡터를 생성하고, 두 문서 내 문장 벡터들 간의 유사도 매트릭스를 구축한다. 그리고 유사도 매트릭스를 다양한 유형의 레이어들을 통해서 단일 벡터로 매핑하고, 소프트 맥스 레이어를 통과하여 두 문서 간의 유사도를 판별한다.

뉴스 데이터 쌍의 관계 판별을 위한 실험에서 네 가지 유사도 모델 중 두 가지 유사도 모델이 기존 문서 표현 접근 방법과 베이스라인 모델보다 높은 성능을 보이는 것을 확인할 수 있었다. 그리고 최근 연구인 그래프 기반 접근 방법 [16]과 비교하면 유사한 성능을 보이지만 더 효율적으로 문서 유사도를 측정하는 것을 확인할 수 있었고, 앙상블 접근 방법으로 두 모델의 다른 특성을 확인할 수 있었다.

2. 관련 연구

2.1 텍스트 표현 기반 접근 방법

텍스트 간의 유사도를 측정하기 위한 텍스트 표현 기반 접근의 연구들은 주로 의역 인식 [6], 질의응답 시스템 [7,8], 정보 검색, 추출 기반 텍스트 요약 등의 자연어처리 분야에서 문장 또는 짧은 텍스트 단위의 유사도를 판별하였다. 이 모델들은 RNN, CNN 등의 모델들을 통해서 문장의 표현을 단일 벡터로 추출하고, 두 벡터 간의 코사인 유사도나 Siamese (symmetric) 뉴럴 네트워크 [14] 등을 통해서 문장 간의 유사도를 측정한다.

[6]의 ARC-I 모델은 입력 텍스트의 지역성을 유지하기 위해 CNN을 통해서 두 개의 문장 벡터를 생성한다. 그리고 두 문장 벡터를 연결하고, 이 벡터를 MLP (multi-layer perceptron)을 통과하여 문장 간의 유사도를 측정한다. 이 모델은 같은 논문의 상호 작용 기반 모델인 ARC-II 보다 일반적으로 낮은 성능을 보였고, 네트워크를 통해서 구축된 문장의 표현의 품질에 따라 성능에 큰 영향을 보였다. [7]은 동일한 CNN으로 생성된 문장 벡터 간의 내적으로, [8]은 BiLSTM으로 생성된 두 문장 벡터 간의 파라미터 추가된 내적으로 두 문장 벡터의 유사도를 계산하고, 그 유사도 스칼라 값을 하나의 자질로 추가한 문장 쌍 벡터를 생성하였다. 이들 연구에서는 대부분 문장 또는 짧은 텍스트의 특성만 고려하여 네트워크를 구성하였고, 이들 네트워크에서 문서에서 나타나는 긴 문맥과 지역적 매칭을 반영하기 어려울 수 있다.

문서 단위의 유사도를 계산하기 위해서는 단어-문장-문서로 이루어진 문서 단위의 계층적 구조를 반영해야 한다. HAN (hierarchical attention network) [5]는 이를 위해 문장 레벨 인코더와 문서 레벨 인코더를 구분하여, 계층적인 네트워크를 구성하였다. 각 인코더는 GRU 기반의 RNN 네트워크와 주의 메커니즘 (attention mechanism)을 같이 사용하여, 문서 분류에서 중요한 단어나 문장에 더 높은 가중치를 부여하도록 한다. Hierarchical Transformer [15]은 주의 기반 네트워크를 Transformer 모델로 대체하여 계층 구조의 네트워크를 구축하였다. 하지만 이 모델들은 문서 분류나 텍스트 요약을 대상으로 연구가 이루어져 단일 텍스트를 대상으로만 연구되었다. 본 논문에서 이 모델 중 HAN을 실험의 베이스라인으로 구성하고, 제안하는 모델에서 일부 차용하여 문서 간의 유사도에서 계층 구조의 효용성을 알아본다.

2.2 상호 작용 기반 접근 방법

상호 작용 기반 접근 방법 [6,9-13]은 표현 기반 접근 방법과 대조적으로 전체 텍스트의 표현을 사용하지 않고, 하위 단계의 표현 간의 상호 작용을 바탕으로 전체 문서의 유사도를 추정하였다. 주로 문장 또는 짧은 텍스트의 유사도를 측정하기 위해서 단어 임베딩 단위의 매칭 매트릭스를 사용하였고, 이 매칭 매트릭스의 패턴이나 함축된 의미로부터 각종 다중 레이어들을 통해서 매칭 매트릭스로부터 효과적으로 유사도를 추정하고자 하였다.

[6]의 ARC-II 모델은 앞의 ARC-I과 달리 두 텍스트 간의 단어 레벨 상호작용을 일반화한 모델로, 단어 벡터 간의 관계로 구성된 2차원 행렬을 CNN의 입력으로 사용한다. 이 모델은 텍스트의 전체적인 의미보다 각 단어의 매칭의 형태로 나타나는 지역성에 초점을 두었다. 실험에서도 ARC-I보다 일반적으로 높은 성능을 보였지만 복잡한 문서 구조나 의미적인 매칭이 나타나는 문제에서 지역성의 효과가 적게 나타나는 것을 확인했다. MatchPyramid [10]는 단어 임베딩 간의 코사인 유사도로 구성된 2차원 행렬을 CNN의 입력으로 사용한다. MV-LSTM [11,12]은 두 문장의 Bi-LSTM의 위치 별 단어 벡터를 활용하여 2차원 상호 작용 행렬을 구성하였고, [11]은 코사인 유사도나 이차 선형 레이어로, [12]는 코사인 유사도, 유클리드 거리와 내적을 같이 조합하여 두 벡터 간의 각도와 차이를 측정하였다. [13]은 GRU 기반 인코더와 두 텍스트를 교차하는 각종 메커니즘을 통해서 유사도를 판별하였다. 이들은 문장의 전체적인 의미보다 각 단어 간의 매칭 패턴에 초점을 두었는데, 이와 같은 지역성에 초점을 둔 네트워크가 문서 쌍의 유사도를 측정하는데 도움이 될 것이라고 생각한다. 왜냐하면 일반적으로 문장보다 긴 시퀀스이며, 주요 토픽과 관련 없는 부차적인 정보도 많이 첨부되고, 이들이 문서의 전체 표현에 불필요하게 포함되기 때문이다. 따라서 이 논문에서 전체적인 문서의 표현보다 문장 간의 매칭 패턴으로 전체 문서 간의 유사도를 측정하고자 한다.

2.3 그 외의 문서 쌍 유사도 접근 방법

최근 문서 쌍 관련 연구에서는 기존의 CNN, RNN, Transformer 등의 딥 뉴럴 네트워크의 모델이 문서 단위의 매우 긴 문맥을 반영하지 못하는 점을 고려하여 그래프 기반 네트워크 [16,17]를 주목하고 있다. [16]은 두 문서 간의 관계를 추정하기 위하여 컨셉 간의 상호 작용 그래프를 구축한다. 이 그래프에서 각 노드는 두 문서

쌍에서 나타나는 키워드들이고, 에지는 키워드들 간의 동시 출현을 나타낸다. 이 그래프를 바탕으로 두 문서에서 구할 수 있는 각종 자질들을 추가하여 그래프 컨볼루션 네트워크 (graph convolution network)에 입력한다. [17]은 그래프 오토 인코더 (graph autoencoder)를 통해서 입력된 키워드 관계 그래프를 특정 벡터에 사상하고, 두 벡터 간의 유사도 행렬로 클러스터링을 수행한다. 이 모델에서 각 노드의 자질로 해당 키워드가 출현한 문장들의 BERT 인코더 [18]의 평균 벡터가 사용되고, 에지는 이 벡터 간의 코사인 유사도로 정의된다. 이들은 문서를 키워드 그래프로 표현된 문장 집합들로 분할하고, 그래프 네트워크를 통해서 전체 문서 간의 유사도를 측정하였다. 본 연구에서는 이들과 같이 문서를 문장 단위로 분할하지만, 그래프가 아닌 문장 간의 상호 작용을 통해서 유사도를 계산한다. 그리고 문서 쌍 판별 실험에서 [16]과의 비교를 통해서 두 모델 간의 차이를 확인한다.

3. 제안하는 문서 쌍 유사도 모델

본 연구에서는 문서 쌍 유사도 모델의 입력은 두 문서 쌍이고, 각 문서는 단어-문장-문서 형태의 계층적인 구조를 가지고 있다. 그리고 출력은 두 문서 쌍의 유사도를 나타내는 0 ~ 1 사이의 스칼라 값이다. 이 모델에서 문장은 인코더를 통해서 문장 표현 벡터로 변환되고, 문장 간 유사도 계산을 통해서 문장 단위 매칭 매트릭스로 변환된다. 그 후 제안하는 네트워크를 통해서 문장 매칭 매트릭스로부터 문장 쌍 유사도를 나타내는 스칼라 값을 추정한다.

3.1 문장 표현을 위한 주의 기반 인코더

제안하는 모델에서 문장의 표현은 HAN [5]에서 제안한 양방향 GRU 네트워크에 주의 메커니즘을 적용한 문장 인코더를 사용한다. 문장 인코더는 문장 표현을 생성할 때 주의 메커니즘을 통해서 중요한 토큰에 높은 가중치를 부여하여 문장 표현력을 높이며, 단어 인코더와 단어 어텐션 모듈로 구성된다.

단어 인코더 x_{it}는 i번째 문장 t번째 단어의 Word2Vec 표현이다.

$$\vec{h}_{it} = \overrightarrow{GRU}(x_{it}), t \in [1, T]$$
$$\overleftarrow{h}_{it} = \overleftarrow{GRU}(x_{it}), t \in [T, 1]$$
$$h_{it} = [\vec{h}_{it}, \overleftarrow{h}_{it}]$$

단어 어텐션 α_{it}는 i번째 문장 t번째 단어의 어텐션 점수, s_i는 i번째 문장의 벡터 표현이다.

$$u_{it} = \tanh(W_w h_{it} + h_w)$$
$$\alpha_{it} = \frac{\exp(u_{it}^\top u_w)}{\sum_t \exp(u_{it}^\top u_w)}$$
$$s_i = \sum_t \alpha_{it} h_{it}$$

3.2 상호 작용 기반 유사도 모델

앞에서 인코더를 통해서 생성된 문장 벡터 표현은 개별 문장 내의 하위 구성요소에만 근거하여 도출된 벡터로, 문서 내에서 문장 간의 의존성 및 맥락 정보가 반영되어 있지 않다. 따라서 연구 [19]에서 제안된 다중 헤드 셀프 어텐션 (multi-head self-attention: $MHSA$) 레이어를 통해서 문장 간의 맥락 정보를 반영한다. 문서 쌍의 모든 문장 벡터가 다음과 같이 입력되고, 여기서 출력된 문장 벡터들을 기반으로 매칭 매트릭스를 생성한다:

$$z_{1:T}^{(i)} = MHSA_{h=8}(s_{1:T}^{(i)}), i \in [1,2]$$

위의 수식에서 $s^{(i)}$는 문서 쌍 내 문서 i의 문장이고, 레이어의 h는 다중 헤드 셀프 어텐션의 헤드 수이다. $z^{(i)}$는 문장 간의 맥락 정보가 반영된 문서 i의 문장 출력 벡터이다. 기존 연구들에서는 입력 텍스트의 각 하위 요소 벡터 (주로, 단어 벡터)들 간의 상호 관계를 두 벡터의 내적이나 코사인 유사도 등을 사용한 2차원 매칭 매트릭스로 표현하였다. 본 연구에서는 생성된 문장 벡터 $z^{(1)}$, $z^{(2)}$로부터, 두 벡터의 거리 값으로 구성된 2차원 매칭 매트릭스와 두 벡터의 거리 벡터로 구성된 3차원 매칭 매트릭스를 모두 사용하여 모델을 구축하였다.

3.2.1 거리 기반 유사도 모델

L개의 h차원 문장 표현 벡터로 구성된 문장 표현 $z^{(1)}$, $z^{(2)}$에 의해 생성되는 2차원 매칭 매트릭스 $M \in \mathbb{R}^{L \times L}$은 아래와 같이 정의된다.

$$M_{i,j} = \exp\left(-\left|\left|z_i^{(1)} - z_j^{(2)}\right|\right|_1\right)$$
$$= \exp\left(-\sum_{k=1}^{h} |z_{i,k}^{(1)} - z_{j,k}^{(2)}|\right), i, j \in [1, L]$$

이후 M을 4개의 CNN - 풀링 레이어에 통과시킨 후 피드포워드 레이어 및 소프트맥스를 적용하여 출력 $o \in \mathbb{R}^2$를 도출한다.

3.2.2 거리 벡터 기반 유사도 모델

L개의 h차원 문장 표현 벡터로 구성된 문장 표현 $z^{(1)}$, $z^{(2)}$에 의해 생성되는 3차원 매칭 매트릭스 $M \in \mathbb{R}^{L \times L \times h}$는 아래와 같이 정의된다.

$$M_{i,j} = |z_i^{(1)} - z_j^{(2)}|$$

기존 연구 [6, 10-12]에서는 CNN과 풀링 레이어를 통

해 입력 매칭 매트릭스로부터 더 작은 크기의 상위 단계 매트릭스를 얻는다. 하지만 기존 연구들이 목적으로 하는 문장 단위 입력의 경우 고착화된 어순이나 빈번한 단어 등으로 인해 유사한 입력 쌍에서 대각 성분이 강조된 유사한 매칭 매트릭스가 나올 가능성이 높다. 반면 문서 입력은 동일한 사건이나 내용을 다루고 있더라도 문장 입력과 같이 문장들이 비슷한 배치를 따르는 것은 아니므로, 공간적 지역성에 근거한 CNN은 적합하지 않다고 판단, CNN 모듈을 어텐션 기반 모듈로 대체하였다. 어텐션 모듈에서 쿼리는 입력 정보들로부터 특정 정보들만을 추출하는 선택자의 역할을 하는 핵심 입력이다. 본 연구에서는 매칭 매트릭스로부터 유의미한 문장 간 거리 정보를 추출하기 위해 학습 가능한 어텐션 쿼리 입력을 사용하며, 쿼리 입력의 재생성 매커니즘을 달리 하여 성능을 평가하였다.

본 연구에서는 3차원 매칭 매트릭스 M을 다중 헤드 어텐션 (multi-head attention) 모듈을 이용해 축소한다. 이를 위해, 입력 매칭 매트릭스보다 작은 어텐션 쿼리 입력 $Q^{(1)} \in \mathbb{R}^{L^{(1)} \times L^{(1)} \times d^{(1)}}, L^{(1)} < L$ 을 독립적인 학습 가능 파라미터로 둔다.

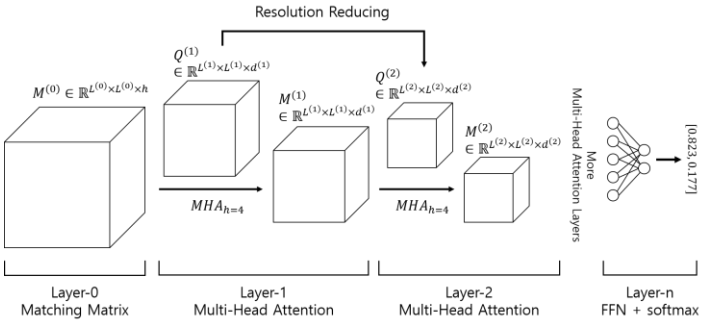

그림 1. 문장 간 매칭 매트릭스를 입력으로 한 문서 간 유사도 모델

이후 M을 다중 헤드 어텐션의 키, 밸류로, $Q^{(1)}$을 쿼리로 입력하여 출력 매트릭스를 얻는다. 이때 출력 매트릭스 $M^{(1)}$은 $Q^{(1)}$와 동일한 형태를 갖는다. 그림 1에서와 같이 위 과정을 반복 실행하여 매칭 매트릭스의 크기를 축소한다. 다중 헤드 어텐션의 내부 구조에 의해, 입력 M의 마지막 차원 성분은 두 개의 선형 레이어에 의해 Q의 마지막 차원 성분과 동일한 크기로 사영되어 어텐션 유닛 내부의 키, 밸류 요소로 사용된다. Q는 입·출력 차원이 동일한 선형 레이어를 통과해 선형 변환된다. 각 선형 레이어의 활성화 함수로는 tanh를 사용하였다.

매 단계마다 다중 헤드 어텐션의 쿼리 입력으로 주어지는 $Q^{(i)}, i \in [1, n-1]$는 크기가 감소한다. 즉, $L^{(i)} < L^{(j)}, d^{(i)} < d^{(j)}, 1 \le i < j < n$을 만족한다. 본 연구에서는 이를 쿼리의 해상도가 줄어드는 과정으로 보고, 이 과정을 아래의 경우로 달리 하여 성능을 확인하였다.

평균 풀링 2차원 평균 풀링과 선형 레이어를 사용한다. 평균 풀링 레이어는 $d^{(i)}$개의 $L^{(i)} \times L^{(i)}$ 입력을 받아 같은 수의 $L^{(i+1)} \times L^{(i+1)}$ 출력을 낸다($Q^{(i)}_{pooled}$). 이를 선형 레이어에 통과시켜 출력의 마지막 차원 성분을 사영함으로써 $L^{(i+1)} < L^{(i)}, d^{(i+1)} < d^{(i)}$를 만족하는 더 작은 크기의 쿼리 $Q^{(i+1)} \in \mathbb{R}^{L^{(i+1)} \times L^{(i+1)} \times d^{(i+1)}}$을 얻는다.

$$Q^{(i)}_{pooled} = AvgPool2d\left(Q^{(i)}\right), Q^{(i)}_{pooled} \in \mathbb{R}^{L^{(i+1)} \times L^{(i+1)} \times d^{(i)}}$$
$$Q^{(i+1)} = \tanh\left(W_p Q^{(i)}_{pooled} + b_p\right)$$

CNN 2차원 CNN 레이어를 사용하였으며, 커널 크기는 $(5,5)$로 두었다. $L^{(i)}$은 H와 W에, $d^{(i)}$는 채널 수에 대응된다. 패딩 크기를 조절하여 출력의 H, W를 $L^{(i+1)}$ 크기로 맞추었다.

변분 추론 기반 재생성 Variational Auto Encoder [20]모델을 기반으로 $d^{(i)}$개의 $L^{(i)} \times L^{(i)}$ 입력을 받아 같은 수의 $L^{(i+1)} \times L^{(i+1)}$ 출력을 낸다($Q^{(i)}_{reduced}$). 이후 평균 풀링의 선형 레이어와 동일한 방법으로 마지막 차원 성분을 사영하여 더 작은 크기의 쿼리 $Q^{(i+1)} \in \mathbb{R}^{L^{(i+1)} \times L^{(i+1)} \times d^{(i+1)}}$을 얻는다. 일반적인 어텐션 매커니즘과 달리 본 연구에서 사용된 어텐션 모듈에서는 입력 데이터가 아닌 직접 학습을 기반으로 쿼리 입력을 생성하므로, 과적합 문제에 취약할 수 있다. 변분 추론 기반 접근은 베이지안 관점에서 확률 분포의 잠재 변수를 명시적으로 모델링하여 변수의 분포를 학습하므로 변수에 대한 강력한 정규화 효과를 기대할 수 있다.

$$Q^{(i)}_{reduced} = VAE\left(Q^{(i)}\right), Q^{(i)}_{reduced} \in \mathbb{R}^{L^{(i+1)} \times L^{(i+1)} \times d^{(i)}}$$
$$Q^{(i+1)} = \tanh\left(W_p Q^{(i)}_{reduced} + b_p\right)$$

독립 쿼리 모든 $Q^{(i)}$를 i에 독립적인 학습 가능 매트릭스로 정의한다.

3.2.3 메타 정보 필터

문서의 메타 정보 필터에 따른 성능 차를 확인하였다. 메타 정보로는 연구에서 사용된 뉴스 기사 데이터셋의 각 문장이 서술하는 대상(사건)의 날짜 정보를 사용하였다. 날짜 필터 $d \in \mathbb{R}^{L \times L}$ 는 아래와 같이 정의된다. $date^{(m)}_k$는 $m \in [1,2]$번째 문서 $k \in [1, L]$번째 문장이 서술하는 대상의 날짜 정보이다. s는 학습 가능한 변수로, 날짜 필터의 활성화 정도를 결정한다.

$$d_{i,j} = sigmoid\left(\left|date^{(1)}_i - date^{(2)}_j\right| + s\right)$$

추가로, 문장 쌍 중 한 문장이라도 패딩된 입력일 경우 $sigmoid(-inf) = 0$로 해당 지점을 마스킹하였다. 날짜 필터를 3차원 매칭 매트릭스 M에 곱하여 문장 간 거리 벡터의 값을 보정한다.

4. 실험

4.1 데이터 구축 및 통계

본 연구에서는 2017년 1월 ~ 2018년 12월까지의 연합뉴스 기사 데이터셋을 활용하였다. 입력된 질의어에 대한 뉴스 검색을 통해서 나타난 검색 결과들을 K-means 클러스터링 방법으로 기사를 쌍으로 구성하여, 아래의 기준에 의해 수작업으로 기사 내용의 동일성을 0(동일하지 않음)과 1(동일)의 두 값으로 태깅했다.

- 문서를 대표하는 주요 사건이 일치
- 주요 사건이 시간적 또는 공간적으로 동일
- 밀접하게 연관된 인물이나 기관 출현

학습 데이터셋 8118쌍과 평가 데이터셋 2000쌍으로 구성되어 있으며, 평가 데이터셋에서 Label이 0인 쌍과 1인 쌍의 비율은 38:62이다. 0-패딩과 절삭을 통해 한 문장에 50단어, 한 단어에 20문장으로 구성하였다. 2017년 ~ 2019년 뉴스 데이터 집합으로 사전 학습된 256 차원의 Word2Vec 모델를 단어 임베딩 벡터로 사용하였다. 학습 과정에서 손실 함수는 Cross Entropy Loss를 사용했고, Adam Optimizer를 사용해 Tesla V100 24GB GPU 상에서 100 에폭 (epoch) 이내로 학습이 이루어졌다.

4.2 베이스라인 모델들

4.2.1. 서포트 벡터 머신 (SVM) 모델

뉴럴 넷을 사용하지 않는 베이스라인 모델로 서포트 벡터 머신 모델을 사용하였다. 모델에서 사용하는 자질로 기사 제목, 작성 날짜, 기사의 각 문장의 벡터 표현 (Word2Vec 단어 표현의 평균)의 코사인 유사도를 사용하였다.

4.2.2. 텍스트 표현 기반 접근

텍스트 표현 기반 모델로 HAN 기반 문서 벡터 비교 모델을 사용했다. $v^{(1)}$, $v^{(2)}$는 두 입력 문서의 HAN 벡터 표현이다. 각 벡터 표현, 두 벡터의 L1 거리, 두 벡터의 코사인 유사도를 연결한 벡터 v를 선형 레이어에 통과시킨 후 소프트맥스 함수를 적용, 출력 $o \in \mathbb{R}^2$를 도출한다.

$$v = concat([v^{(1)}, v^{(2)}, |v^{(1)} - v^{(2)}|, \frac{v^{(1)} \cdot v^{(2)}}{||v^{(1)}|| ||v^{(2)}||}])$$

$$o = softmax(W_c v + b_c), o \in \mathbb{R}^2$$

4.3 모델 성능 평가

표 1은 상술한 조건 하에서 대상 모델의 성능을 세 번 실험한 결과의 평균값이다. 거리 기반 CNN 모델은 (5×5 커널 - 최대 풀링)×2 - (3×3 커널 - 최대 풀링)×2의 구조이다.

거리 벡터 기반 모델의 어텐션 출력 크기 축소는 표 2의 과정에 따라 이루어졌다.

텍스트 표현 기반 접근인 HAN 기반 모델은 기사 제목, 작성 날짜, 기사의 각 문장의 벡터 표현 (word2vec 단어 표현의 평균)의 코사인 유사도를 자질로 사용한 SVM 기반 모델보다 낮은 성능을 보였다. 텍스트 표현 기반 모델은 문서의 의미 정보를 하나의 벡터에 풍부하게 담는 것이 관건인데, 학습 데이터가 적고 (8118쌍) 두 문서를 비교하기까지의 높은 복잡도에 비해 학습의 최종 레이블 형태는 다중 클래스가 아닌 문서의 유사성 여부만으로 구성되므로, 문서의 의미적 정보를 이끌어냄에 있어 어려움을 겪은 것으로 보인다. 또한 병렬화할 수 없는 RNN 연산으로 구성되어 있어 아이템 당 연산 시간도 CNN, 어텐션 등 병렬 가능 연산으로 구성된 상호 작용 기반 접근에 비해 길다.

상호 작용 기반 접근 중 문장 쌍 간의 관계를 하나의 값으로 표현하는 거리 기반 모델은 이를 벡터로 표현하는 거리 벡터 기반 모델보다 성능이 다소 떨어졌다. 이전 여러 연구에서 다루었던 거리 기반 방법이 텍스트의 두 하위 요소 간의 거리를 과하게 단순화시켰다는 추측을 해볼 수 있다.

어텐션 쿼리의 축소 과정에서 쿼리의 공간적 지역성에 의지한 방법들 - 풀링, 2차원 CNN - 은 그렇지 않은 모델들보다 낮은 성능을 보였다. 이전 연구들에서 수행한 문장 단위 입력에서 어순이나 문장구조의 유사성 등에 의해 유사한 공간적 매칭 패턴이 입력 간의 관계에 유의미한 영향을 주었던 것에 비해, 문서 단위 입력에서는 상대적으로 하위 구성 요소로 사용한 문장 자체가 이미 일정 수준의 완결된 의미를 가지므로 이들의 배치가 실제 문서의 유사성에 큰 요인이 아니었기 때문으로 분석된다. 또한 어텐션 쿼리의 변분 추론 기반 재생성 유닛을 사용

표 1: 뉴스 기사 매칭 성능 검증 결과

방법	모델	정확도(%)	시간(ms/it)
자질 기반	SVM	81.00	-
표현 기반	HAN based	77.75	16.21
상호 작용 기반	Value based - CNN	74.50	15.04
	Vector based - AvgPool2D	79.70	16.60
	Vector based - Conv2D	77.50	14.10
	Vector based - VI	**82.80**	14.30
	Vector based - Indep.	82.13	**11.72**

표 2: 거리 벡터 기반 모델의 어텐션 출력 Shape 축소 과정

	M shape	Q shape
1	$20 \times 20 \times 256$	$12 \times 12 \times 256$
2	$12 \times 12 \times 256$	$8 \times 8 \times 32$
3	$8 \times 8 \times 32$	$2 \times 2 \times 16$

한 모델의 성능이 독립적인 쿼리를 사용한 모델보다 좋았는데, 이는 변분 추론이 임의 노이즈 첨가의 과정을 포함하기 때문에 데이터가 부족한 상황에서도 주어진 데이터로 일반화된 쿼리 패턴을 만들기 유리했던 것으로 분석된다.

4.4 메타 정보 필터의 효과

표 3: VI 모델에서 Date Filter의 유무에 따른 성능

모델	정확도(%)	시간(ms/it)
Vector based - VI	82.80	14.30
+ Date Filter	83.80	14.88

메타정보로 사용한 문장 날짜 정보로 생성된 필터가 성능에 영향을 주는지 확인하였다. 표 3에서 볼 수 있듯, 비교모델 상에서 가장 좋은 성능을 보여준 거리 벡터 기반 - VI 모델에 날짜 필터를 적용한 결과 1%p가량의 성능 향상이 있었다. 태스크마다 이용 가능한 메타정보가 다르므로 본 결과의 성능 향상을 일반화할 수는 없지만, 사전 학습된 언어 모델 (language model)을 사용하지 않는 텍스트 유사도 측정 분야에서 부족한 정보량을 메타정보의 활용으로 극복할 수 있음을 확인하였다.

4.5 그래프 기반 모델과의 차이

본 논문에서 제시한 모델은 문장 단위의 의미적 매칭 패턴을 중심으로 두 입력 문서의 유사도를 도출하였다. 이는 키워드 단위의 의미적 매칭을 기반으로 유사도를 도출하는 [16]의 방식과 대조적이다. 두 모델의 성능 및 오류 분석 결과는 표 4와 같다. 본 논문에서 제시한 모들을 IB(Interaction Based) 모델로 기재하였다.

표 4. 각 모델의 성능 및 오류 분석 결과. [1]: 자질 추출에 기반한 SVM 모델, [2]: [16], [3]: 본 논문에서 제시한 Vector based - VI + Date Filter 모델, [4]: 각 모델의 앙상블 모델. 최종 학습된 모델들의 예측값 평균치를 사용했다.

모델	정확도(%)	제1종 오류(개)	제2종 오류(개)
SVM[1]	81.00	174	191
GCN[2]	84.10	134	184
IB[3]	83.80	161	163
GCN+IB[4]	84.70	142	164
SVM+GCN+IB[4]	85.45	128	163

성능의 경우 GCN 기반 모델과 IB 모델이 거의 유사했다. IB모델의 경우 단어 벡터 입력으로부터 유사도 출력까지의 과정이 End-to-End 학습으로 이루어진다. 하지만 GCN 기반 모델은 End-to-End로 학습할 경우 성능이 잘 나오기 어렵다. 학습과 추론 과정에서 문서의 메타정보 및 임베딩 정보들을 종합하여 문서로부터 키워드 그래프를 구축해야 하고, 이 과정의 복잡도가 매우 높아 많은 시간이 소요된다(학습 데이터 8K 문장 쌍에서 4시간 정도 소요). 하지만 IB 모델은 학습 이전 자질 추출에 의한 추가 자원 소모가 없으므로 유사한 성능을 보이지만 효율성 측면에서 더 우수하다.

베이스라인 모델인 SVM에 비해 GCN 기반 모델 [16]은 제1종 오류를, IB 모델은 제2종 오류를 크게 줄인 것으로 나타났다. GCN 기반 모델 [16]은 키워드 중심의 비교를 통해 유사한 단어가 등장하지만 다른 내용을 담은 문서 쌍들을 명확히 구분하였다. 서로 다르지만 연속적으로 발생한 두 사건을 다루는 문서 쌍에서, 다른 모델들은 비슷한 단어들이 여러 번 등장하고 이전 사건들에 대한 서술이 반복됨에 따라 분류에 어려움을 겪은 반면, GCN 기반 모델 [16]은 각 인물과 키워드를 뚜렷이 나누어 명확히 분류했다. IB 모델은 문장 단위의 비교를 통해 서술 순서만 다르고 주제가 같은 문서 쌍들을 명확히 구분하였다. 특히 문서 내 문장들이 문서 전체에 있어 의미상 비교적 독립적이고, 입력 쌍에서 문장 간 의미 대응이 명확한 경우에 강점을 보였다. GCN 기반 모델 [16]은 키워드 별 문장 인코딩과 임베딩 추출 과정이 단층 CNN으로 이루어져 깊은 맥락 정보를 반영하기 어렵다는 점, IB 모델은 문장의 위치 정보가 손실된다는 점이 개선의 여지로 남았다.

앙상블은 이러한 단점들을 보완해주었다. IB 모델의 위치 정보 손실은 키워드 별 비교를 통해 보완하였고, GCN 기반 모델 [16]의 문장 인코딩 문제는 IB 모델의 반복적인 어텐션 메커니즘으로 보완하였다. 결과적으로 GCN+IB 모델은 제1/2종 오류를 모두 크게 줄인 것으로 나타났으며, 추출한 자질을 예측결과에 직접 반영한 SVM+GCN+IB 모델은 본 태스크에서 가장 높은 정확도를 기록했다.

5. 결론

본 연구에서는 과거 문장 단위 텍스트 유사도 매칭 연구들의 두 가지 접근인 텍스트 표현 기반, 상호 작용 기반 접근을 문서 단위 텍스트 입력에의 적용 가능성을 중심으로 분석했다. 문서 단위 입력에 대응하기 위해 기존 CNN에 의존했던 상호 작용 기반 모델들의 핵심 모듈을 어텐션 모듈로 대체한 새로운 모델을 제시했다. 독립적인 어텐션 쿼리 인자가 매칭 매트릭스의 일정한 패턴을 공간적 지역성이 아닌 내용 자체의 유사성에 의존하여 추출함으로써 문서 단위 입력에서의 성능을 향상했다. 이 과정에서 핵심 프로세스인 어텐션 쿼리 해상도 축소에 풀링, CNN, 변분 추론의 기법을 적용하여 성능을 분석하였으며, 데이터 부족 상황에서 입력 데이터의 메타정보를 필터의 형태로 활용할 수 있는 방법을 적용하였다.

또한 본 모델과 접근 방식이 대조적인 GCN 기반 모델 [16]을 대상으로 오류분석을 통해 접근 방식의 차이가 다른 예측 양상을 도출할 수 있음을 확인하였다. 두 모델의 앙상블을 통해 높은 성능을 달성했으며, 대조적인 접근 방식이 갖는 상승 효과를 확인하였다.

감사의 글 본 연구는 엔씨소프트 산학연구용역 과제의 지원을 받아 수행되었음.

참고문헌

[1] Charu. C. Aggarwal and ChengXiang. Zhai, “A Survey of Text Clustering Algorithms”, *Mining text data*, Springer, Boston, MA, pp. 77-128, 2012.

[2] Mikhail Bilenko, Sugato Basu and Raymond J. Mooney, “Integrating Constraints and Metric Learning in Semi-Supervised Clustering”, In *Proceedings of the twenty-first international conference on Machine learning*, 2004.

[3] John S. Whissell and Charles L. A. Clarke, "Improving document clustering using Okapi BM25 feature weighting", *Information Retrieval*, 14, pp. 266-487, 2011.

[4] Yoon Kim, “Convolutional Neural Networks for Sentence Classification”, In *EMNLP-14*, 2014.

[5] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola and Eduard Hovy, “Hierarchical Attention Networks for Document Classification”, In *Proceedings of NAACL-HLT 2016*, pp. 1480-1489, San Diego, California, June 12-17, 2016.

[6] Baotian Hu, Zhengdong Lu, Hang Li and Qingcai Chen, “Convolutional Neural Network Architectures for Matching Natural Language Sentences”, In *Advances in Neural Information Processing Systems*, pp. 2042-2050, 2014.

[7] Kateryna Tymoshenko, Daniele Bonadiman and Alessandro Moschitti, “Convolutional Neural Networks vs. Convolutional Kernels: Feature Engineering for Answer Sentence Reranking”, In *Proceedings of NAACL-HLT 2016*, pp. 1268-1278, San Diego, California, June 12-17, 2016.

[8] Yi Tay, Minh C. Phan, Luu Anh Tuan and Siu Cheung Hui, “Learning to Rank Question Answer Pairs with Holographic Dual LSTM Architecture”, In *Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval*, pp. 695-704, 2017.

[9] Zhengdong Lu and Hang Li, “A Deep Architecture for Matching Short Texts”, In *Advances in Neural Information Processing Systems*, pp. 1367-1375, 2013.

[10] Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Shengxian Wan and Xueqi Cheng, “Text Matching as Image Recognition”, In Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI-16), 2016.

[11] Shengxian Wan, Yanyan Lan, Jiafeng Guo, Jun Xu, Liang Pang and Xueqi Cheng, “A Deep Architecture for Semantic Matching with Multiple Positional Sentence Representations”, In *Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI-16)*, 2016.

[12] Hua He and Jimmy Lin, “Pairwise Word Interaction Modeling with Deep Neural Networks for Semantic Similarity Measurement”, In *Proceedings of NAACL-HLT 2016*, pp. 937-948, 2016.

[13] Chuanqi Tan, Furu Wei, Whenhui Wang, Weifeng Lv and Ming Zhou, “Multiway Attention Networks for Modeling Sentence Pairs”, In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI-18)*, 2018.

[14] Bromley, Jane and Guyon, Isabelle and LeCun, Yann and Sackinger, Eduard and Shah, Roopak, “Signature verification using a" siamese" time delay neural network”, In *Proceedings of Advances in neural information processing systems*, pp.737-744, 1994.

[15] Yang Liu and Mirella Lapata, “Hierarchical Transformers for Multi-Document Summarization”, In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 5070-5081, Florence, Italy, July 28, 2019.

[16] Bang Liu, Di Niu, Haojie Wei, Jinghong Lin, Yancheng He, Kunfeng Lai and Yu Xu, “Matching Article Pairs with Graphical Decomposition and Convolutions”, In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 6284-6294, Florence, Italy, July 28, 2019.

[17] Billy Chiu, Sunil Kumar Sahu, Derek Thomas, Neha Sengupta and Mohammady Mahdy, "Autoencoding Keyword Correlation Graph for Document Clustering", In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp.3974-3981, 2020.

[18] Jacob Devlin, Ming-Wei Chang, Kenton Lee and Kristina Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”, *arXiv preprint arXiv:1810.04805*, 2018.

[19] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser and Illia Polosukhin, “Attention Is All You Need”, In *Proceedings of 31st Conference on Neural Information Processing Systems (NIPS 2017)*, Long Beach, CA, USA, 2017.

[20] Kingma, Diederik P and Welling, Max, “Auto-encoding variational bayes”, *arXiv preprint arXiv:1312.6114*, 2013.