**Abstract**

우리는 일정치 않은 길이의 시퀀스들의 쌍들로 구성된 레이블이 매겨진 데이터셋을 처리하기 위해 Long Short-Term Memory(LSTM) 네트워크와 유사한 방식을 제시합니다. 우리의 모델은 문장들 간의 의미적 유사성을 평가하기 위해 적용되며, 신중하게 수작업으로 작성된 feature들과 최근에 제안된 아주 복잡한(어려운) 신경망 시스템보다 우수합니다. 이러한 용도로 사용하기 위해, 우리는 워드 임베딩 벡터를 synonymic한 정보와 함께 LSTMs에 제공합니다. 워드 임베딩 벡터는 특정 단어, 문법과는 상관없이 문장 안에서 표현되는 근본적인 의미를 담기 위해 고정된 크기의 벡터를 사용합니다. 후속 작업을 맨해튼 거리에 의존하도록 제한함으로써, 우리의 모델로 학습된 문장의 표현(표상)이 복잡한 의미 관계를 반영하는 잘 구조화된 공간을 형성하도록 한다. (예를 들어 평면 위의 맨해튼 거리가 (p1, p2)과 (q1, q2) ( p 1 , p 2 ) {\displaystyle (p\_{1},p\_{2})} ( q 1 , q 2 ) {\displaystyle (q\_{1},q\_{2})}사이이면 |p1 – q1| + |p2 – q2|이다| p 1 − q 1 | + | p 2 − q 2 | {\displaystyle |p\_{1}-q\_{1}|+|p\_{2}-q\_{2}|} .) 우리의 결과는 LSTMs가 복잡한 이해를 요구하는 작업을 수행할 수 있는 강력한 언어 모델이라고 밝혀진 결과들 중 가장 최근의 결과입니다.

**Introduction**

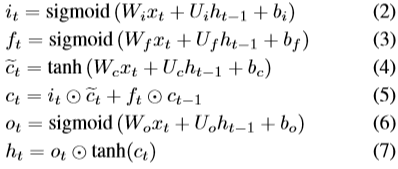
텍스트 이해와 정보 검색은 문장들 사이의 근본적인 의미 유사성을 모델링 함으로써 크게 향상될 수 있는 중요한 작업이다. 특히 좋은 모델은 같은 생각을 표현하는 데 사용되는 단어/구문의 다양성에 취약하지 않아야 한다. 따라서 의미 텍스트 유사성 지표를 학습함으로써 상당한 연구 관심을 유발했다(Marelli et al. 2014). 그러나 이는 어려운 문제로 남아있는데, 왜냐하면 레이블이 매겨진 데이터가 부족하고, 문장은 길이가 일정치 않으면서 복잡한 구조를 가지고 있고, (자연어 처리에서는 지배적이지만) 문장에 내재된 용어의 전문성(전문용어)에 의해 bag-of-words와 tf-IDF모델이 제한되기 때문이다. (c.f. Mihalcea, Corley, and Strapparava 2006).

이러한 아이디어의 대안으로 Mikolov et al.(2013)과 그 외 다른 사람들은 유추와 다른 NLP과제에 대한 신경 언어 표현의 효과성을 보여주었다. 최근에는, 각각의 단어 수준을 넘어 문장과 같은 더 큰 단계로 이러한 아이디어들이 확장되며 관심이 옮겨갔다. 이 아이디어에서는 맵핑이 각 문장을 표현하는 데에 고정된 길이의 벡터를 사용하는 것을 학습한다. (Kiros et al. 2015; Tai, Socher, and Manning 2015; Le and Mikolov 2014).

RNN, 특히 LSTM모델(Hochreiter and Schmidhuber, 1997)이 문장과 같은 길이가 일정치 않은 입력 데이터에 적합하여서, 텍스트 분류(Graves 2012)와 언어 번역(Sutskever, Vinyals, and Le 2014)과 같은 과제에 있어서 성공적이었다. RNNs는 표준 피드-포워드 신경망을 시퀀스 데이터 (x1, … , xT)에 적용시킨다. (1, 2, … , T)에 속하는 t에 대해 은닉층 벡터 ht가 다음식을 통해 계산된다.

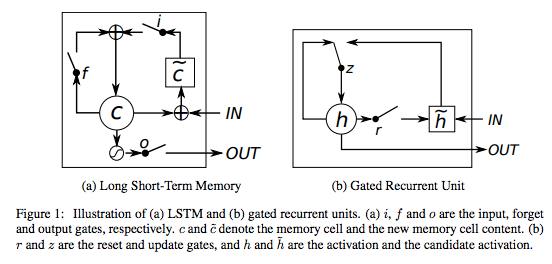


Siegelmann과 Sontag(1995)는 기본 RNN이 ‘튜링 완전’이라는 것을 보여 주었지만, 가중치 행렬의 최적화는 그것의 역전파된 그래디언트들이 긴 시퀀스에 의해 조금씩 증발하기 때문에 어렵다. (튜링 완전이란 어떤 프로그래밍 언어나 추상 기계가 튜링 기계와 동일한 계산 능력을 가진다는 의미이다. 이것은 튜링 기계로 풀 수 있는 문제, 즉 계산적인 문제를 그 프로그래밍 언어나 추상 기계로 풀 수 있다는 의미이다.) 실제로 LSTM은 긴 입력 시퀀스가 들어간 정보를 저장 및 접근할 수 있는 메모리 셀 단위를 사용하여 장거리 의존성을 학습하는 것에 있어서 기본 RNN보다 우수하다. RNN과 마찬가지로 LSTM도 은닉 상태의 표현(표상)을 순차적으로 업데이트하지만, 이러한 단계는 네 가지 구성 요소(실제 값 벡터)가 포함된 메모리 셀에도 의존합니다. 네 가지 구성 요소에는 메모리 상태 ct, 어떻게 메모리 상태 ct가 다른 유닛들에 영향을 주는지를 결정하는 출력게이트 ot, 각각의 새로운 입력과 현재 상태에 기반하여 메모리에서 저장되거나 없어지는 것을 통제하는 역할의 입력게이트 it와 망각게이트 ft가 있다. 아래의 식들은 가중치 행렬과 편향 벡터를 통해 업데이트를 하는 식들이다.



LSTM과 그와 비슷한 다양한 모델들에 대한 자세한 설명은 Graves(2012) 와 Greff et al. (2015)이 설명한다.

LSTM언어 모델의 성공은 현재의 이론적 이해를 이끌어 낼 수 없지만, Sutskever, Vinyals및 Le(2014)는 효과적으로 훈련된 네트워크가 텍스트에 표현된 근본적인 의미를 담는 고정 길이 벡터에 각각의 문장을 매핑한다는 직관을 경험적으로 입증하였다. 최근의 연구들은 Cho et al(2014년)의 단순한 반복 실험 단위(GRU)와 같은 많은 다른 LSTM변형을 제안했다. 광범위한 경험적 분석 후 Greff et al.(2015)등의 사람들은 제안된 변화들 중 어떤 것이 기본 LSTM모델을 능가하는지 여부에 대해 의문이다. 또한 순차적 RNN의 보다 폭넓은 아키텍처 적인 개정판이 양방향, 다중 계층(Graves2012)및 재귀 트리 구조(Socher2014)를 포함한 텍스트 모델링에 도입되었다. 아래는 LSTM과 GRU에 관한 그림이다.



이 연구에서는 충분한 데이터가 주어질 경우 단순한 LSTM 모델이 풍부한 의미를 담은 문장 표현들로 구성된 고도로 구조화된 공간을 학습할 수 있도록 쌍으로 구성된 예제를 통해 학습 교육한다. Sutskever, Vinyals, Le(2014년)의 언어 번역 실험과 같은 우리의 결과는 표준 LSTMs가 복잡해 보이는 NLP문제를 상당히 잘 처리할 수 있음을 보여줍니다. 단순함에도 불구하고, 우리의 접근 방식은 문장들 간의 유사성을 평가하는 현재의 첨단 기술보다 우수한 성능을 보여준다.

공식적으로, 우리는 각 훈련 예제가 고정 길이 벡터(C:\Users\강범서\Desktop\캡처1.PNG)의 한 쌍의 시퀀스(x1(a), … , xT(a)(a)), (x1(b), … , xT(b)(b))와 그 쌍에 해당하는 1개의 라벨 y로 구성되는 감독 학습 환경을 고려한다. 시퀀스 길이는 여러가지일 수 있으며 시퀀스 길이는 예시마다 다를 수 있습니다. 우리가 두개의 시퀀스를 대칭적으로 다루는 동안, 우리의 방식은 쉽게 다음의 케이스로 확장될 수 있습니다. (x1(a), … , xT(a)(a))시퀀스가 하나의 도메인에서 생겨나고 (x1(b), … , xT(b)(b))시퀀스는 또다른 도메인에서 생겨나는 케이스이다. 주어진 y값이 유사성의 근본적인 측정 값을 반영한다고 가정하면, 우리의 알고리즘은 길이가 일정하지 않은 시퀀스의 일반적인 공간에서 이해할 수 있는 고정된 차원수의 구조화된 거리 공간으로의 매핑을 생성합니다. (다차원 스케일링과 같은 매니폴드 내장 기법과 달리 데이터에 없는 새로운 예에 적용할 수 있음)

우리에게 동기를 부여하는 예시는 인간에 의해 의미적 유사성이 y로 레이블 된 쌍들의 예시가 주어지면, 이 예시에서의 문장들 간의 유사성에 점수를 매기는 일이다. 이 경우 각 xi(a)는 첫번째 문장에서의 단어의 벡터 표현을 나타내고 xj(b)는 두번째 문장에서의 단어의 벡터 표현을 나타낸다. 따라서, 우리는 의미를 반영하는 행렬을 학습한다는 명시적인 목표를 위해 LSTMs를 적용하며, Sutskever, Viosals및 Le(2014)의 작업과는 대조적이다.

**Related Work**

의미 유사성 평가는 다양한 용도에 걸쳐 중요하기 때문에, SemEval(2014년)의 첫번째 과제로 선정되었습니다. 여기서 수많은 연구자들이 레이블이 붙은 데이터셋(작문 방식을 포함하는 문장들(SICK)의 쌍을 포함하는)에 방법들을 적용하였다(Marelli et al. 2014). 이 데이터에 대한 경쟁적인 방법은 여러 다른 종류의 feature(예: 단어 오버랩/유사성, 부정 모델링, 문장/구문 구성)와 외부 자원(예: Wordnet)을 모두 활용했습니다. 그리고 다양한 학습 알고리즘이 적용되었습니다. (예: SVM, 랜덤 포레스트, k-최근접 이웃 알고리즘, 모델 앙상블)

Zhao, Zhu, Lan(2014)의 가장 우수한 원본 제출물 중 하나는 잠재적인 의미 분석과 기타 수작업으로 이루어진 여러 feature들을 사용하여 벡터 공간 표현(표상)을 학습합니다. 또 다른 고성능 시스템은 Bjerva et al.(2014)에서 비롯된 것이며, 인기있는 word2vec 신경망 언어 모델(Mikolov et al. 2013)로부터 나온 feature들과 결합하여 공식적인 의미와 논리적인 추론을 대중적인 사용합니다. 우리 모델에서 사용하는 유일한 입력 자료는 word2vec의 워드 벡터입니다.

최근에 들어, 우리의 접근 방식과 더욱 비슷한 세 가지 신경망 방식이 성능적인 부분에서 유의미한 발전을 보여주었습니다. He, Gimpel, Lin(2015)는 정교한 합성곱 신경망(ConvNet)의 변형판은 다양한 규모에서의 많은 컨볼루션에 걸친 다양한 차이점들을 통합함으로써 문장의 유사성을 추론합니다. 이 사람들은 상당한 양의 아키텍쳐적인 엔지니어링이 필요하다고 설명했습니다. 왜냐하면 레이블된 데이터의 제한적인 이용가능성 때문입니다. 우리가 하는 연구작업에서는 훈련 세트를 증가시킴으로써 처리할 수 있는 문제입니다.

Kiros et al.(2015)은 skip-thoughts 모델을 제안하였습니다. 이 모델은 word2vec의 skip-gram 접근법을 단어 레벨에서 문장 레벨까지 확장시킨 것입니다. 이 모델은 각 문장을 GRU의 동작과 함께 RNN 인코더-디코더에 넣습니다. 인코더-디코더는 이 문장의 바로 앞뒤 문장들을 재구성하는 역할입니다. 그들의 접근 방식을 문장의 유사성 과제에 적용시키기 위해, Kiros et al.은 skip-thought벡터를 얻기 위해 첫번째 문장을 RNN 인코더에 통과시킵니다. (인코더의 가중치는 첫 말뭉치들을 통해 훈련한 이후에 고정됩니다.) 그 뒤에는, 따로 분리된 분류기가 각 훈련 예시에 등장하는 문장들의 쌍의 skip-thought벡터들 사이의 차이점과 생성물로부터 얻은 feature를 사용하는 SICK 데이터를 통해 훈련됩니다. Sutskever, Vinyals, Le(2014)의 프레임워크 인코더-디코더 에서와 같이, 의미 속성들은 skip-thought 표현(표상)에서 목표에 대해 명시적으로 타겟이 되기 보다는 간접적인 효과로써 나타납니다.

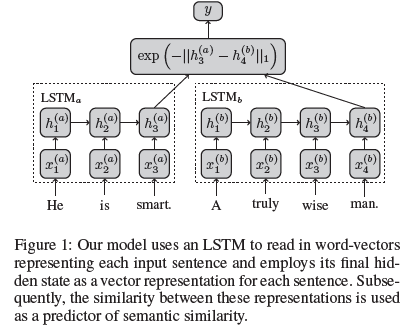
Tai, Socher, Manning(2015)은 표준 LSTMs의 순서에 민감한 연쇄 구조를 트리 구조의 네트워크 토폴로지로 일반화하는 Tree-LSTMs를 제안합니다. 각 문장은 처음에 파스 트리로 변환되며(별도로 훈련을 받은 구문 분석기를 사용하여), Tree-LSTMs는 주어진 단어의 트리 노드의 은닉 상태 및 모든 자식 노드들의 은닉 상태를 구성합니다. 모든 하위 노드 중에서 선택할 수 있습니다. 그 희망은 문장의 구문적 특성을 반영함으로써 파스 트리 구조의 네트워크가 필요한 정보를 순차적으로 제한된 아키텍쳐보다 더 효율적으로 전파할 수 있게 되는 것입니다. 이 모델은 Kiros et al. 와 마찬가지로 문장 유사성에 대해 쓰인다. 여기서 입력 문장의 표현(표상)은 skip-thoughts보다는 Tree-LSTMs에 의해 만들어진다.

우리가 제안한 모델은 또한 큰 말뭉치로부터 별도로 학습된 단어 벡터가 입력값인 신경망을 사용하여 문장을 나타낸다. 그러나, 우리의 목표를 학습하는 접근법은 Kiros et al.의 접근법과는 다르게 직접적으로 주어진 의미의 유사성 레이블을 반영합니다. 그러나 앞서 언급한 신경망들은 문장 표현으로부터 의미의 유사성을 예측하기 위해 복잡한 learner를 사용하지만, 우리는 다음과 같은 더 강력한걸 원합니다. 다시 말해, 의미 구조적인 표현 공간은 간단한 metrics가 문장의 유사성을 잡아내는데 충분할만큼 학습이 되어야 합니다.

이 관점은 또한 얼굴인식에 쓰이는 Siamese 아키텍쳐를 기저로 한다. (Chopra, Hadsell, LeCun(2005)) 이는 대칭적 ConvNets을 이용하며 우리가 여기에서 LSTMs를 사용한다. Siamese 신경망은 많은 양의 metric 학습 과제를 위해 제안되어 왔지만, 우리의 지식선에서는, recurrent 연결은 대부분 발굴되지 않은 채 남아있다.

**Manhattan LSTM Model**

제안된 맨해튼 LSTM모델(MaLSTM)은 그림1에 간략히 묘사되있다. 두개의 네트워크 LSTMa, LSTMb가 있으며 각각 주어진 쌍에서의 문장들 중 하나를 처리하지만, 우리는 이 연구에서는 LSTMa와 LSTMb가 같도록 엮인 가중치를 가진 Siamese 아키텍쳐에만 집중할 것이다. 그럼에도 불구하고, 이 모델의 엮이지 않은 일반적인 버전은 정보 검색과 같은 비대칭적인 도메인에서의 적용에서 더욱 유용할 것입니다.



LSTM은 din차원의 벡터들의 길이가 일정치 않은 시퀀스들의 공간으로부터 Rd(rep)으로의 매핑을 학습한다. (이 연구에서는 din = 300, drep = 50이다.) 좀 더 구체적으로는, (워드 벡터의 시퀀스로 표현되는) 각각의 문장 x1, … , xT는 LSTM으로 통과되어, (2)-(7)의 방정식들을 통해 각 시퀀스 인덱스의 은닉층을 업데이트 시킨다. 문장의 마지막은 hT로 표현되며, 이는 모델의 마지막 은닉층이다. 주어진 문장들의 쌍에 대해, 우리의 접근법은 바로 미리 정의된 유사성에 관련된 함수를 적용시키는 것입니다.

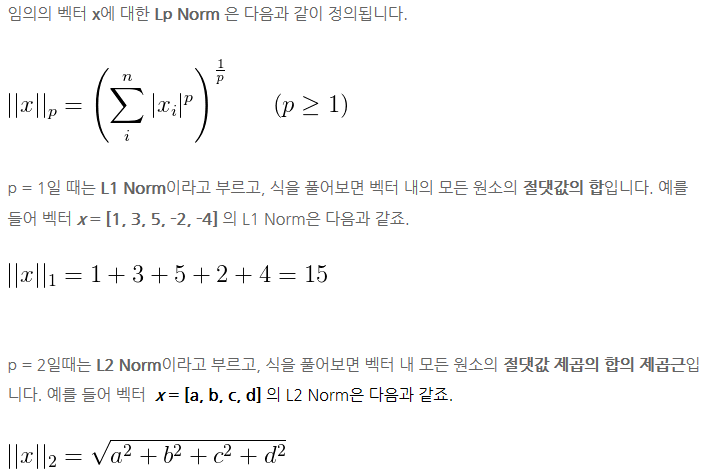
C:\Users\강범서\Desktop\캡처3.PNG를 LSTM 표현물에 적용시킵니다. 표현 공간에서의 유사성들은 나중에 문장들의 근본적인 의미의 유사성을 추론하는데 사용됩니다.

이것을 명심해야 합니다. 이전 텍스트를 토대로 다음 단어를 예측하는 전형적인 RNN 언어 모델과는 다르게, 우리의 LSTMs은 Sutskever, Vinyals, Le(2014)의 인코더처럼 단순하게 동작한다는 것을 명심해야 합니다. 그래서, 훈련 중 역전파되는 하나의 에러 신호는 문장의 표현(표상) hT(a)와 hT(b)사이의 유사성으로부터 나오는 것이며, 이것이 어떻게 유사성을 측정하는 지는 사람이 주석을 단 실측 자료 유사성(ground truth relatedness)와는 다르다. 우리는 다음 식을 사용합니다.

C:\Users\강범서\Desktop\캡처4.PNG

이는 LSTM이 훈련 중 의미의 차이점을 완전히 잡아내주도록 만듭니다. 단점을 보완하기 위해 더 복잡한 learner를 통해 RNN을 보충하는 것보다 낫습니다.

아래 그림을 참고하세요.



Chopra, Hadsell, LeCun(2005)이 지적한 것처럼, 유사성 함수에서 l1을 사용하는 것 보다 l2를 사용하는 것이 전체적인 목표 함수에서 원치 않은 고원(안정기)(plateaus)로 이끌 수 있습니다. 이는 왜냐하면 l2 기반의 모델은 훈련 초기 때 에러를 수정하는 것이 불가능하며, 이때 모델은 의미적으로 다른 문장들이 가깝다고 잘못 믿게 됩니다. 이는 유클리디언 거리의 그래디언트 증발 현상 때문입니다. 경험적으로, 우리가 내는 결과는 우리의 단순한 유사성 함수의 다양한 버전에 걸쳐 안정적입니다. 그러나 우린 맨해튼 거리를 이용하는 함수 g가 코사인 유사도와 같은 다른 합리적인 대안들보다 성능이 조금 더 좋다고 발견하였습니다. (Yih et al. 2011에 쓰임)

**Semantic relatedness scoring**

ㄴㄴ