신경망을 사용한 시퀀스 간 학습

일리아 수츠케버
Google
ilyasu@google.com

오리올 빈얄스
Google
vinyals@google.com

Quoc V. Le
Google
qvl@google.com

초록

심층 신경망(DNN)은 어려운 학습 작업에서 탁월한 성능을 발휘하는 강력한 모델입니다. DNN은 레이블이 지정된 대규모 훈련 집합을 사용할 수 있을 때는 잘 작동하지만, 시퀀스와 시퀀스를 매핑하는 데는 사용할 수 없습니다. 이 백서에서는 시퀀스 구조에 대한 가정을 최소화하는 시퀀스 학습에 대한 일반적인 엔드투엔드 접근 방식을 제시합니다. 이 방법은 다층 구조의 장단기 메모리(LSTM)를 사용해 입력 시퀀스를 고정된 차원의 벡터에 매핑한 다음, 또 다른 심층 LSTM을 사용해 벡터에서 목표 시퀀스를 디코딩합니다. 주요 결과는 WMT'14 데이터 세트의 영어에서 프랑스어로의 번역 작업에서 LSTM이 생성한 번역이 전체 테스트 세트에서 34.8의 BLEU 점수를 획득했으며, 어휘를 벗어난 단어에 대해서는 LSTM의 BLEU 점수가 감점되었다는 점입니다. 또한 LSTM은 긴 문장에서도 어려움을 겪지 않았습니다. 비교를 위해, 구문 기반 SMT 시스템은 동일한 데이터 세트에서 33.3의 BLEU 점수를 얻었습니다. 앞서 언급한 SMT 시스템에서 생성된 1000개의 가설의 순위를 다시 매기는 데 LSTM을 사용했을 때, BLEU 점수는 36.5로 증가하여 이 작업에서 이전 최고 결과에 근접했습니다. 또한 LSTM은 어순에 민감하고 능동태와 수동태에 상대적으로 변하지 않는 합리적인 구문 및 문장 표현을 학습했습니다. 마지막으로, 모든 소스 문장에서 단어의 순서를 바꾸면(대상 문장은 제외) LSTM의 성능이 현저히 향상되는 것을 발견했는데, 이는 소스 문장과 대상 문장 사이에 많은 단기 종속성을 도입하여 최적화 문제를 더 쉽게 만들었기 때문입니다.

1 소개

심층 신경망(DNN)은 음성 인식[13, 7] 및 시각적 객체 인식[19, 6, 21, 20]과 같은 어려운 문제에서 탁월한 성능을 발휘하는 매우 강력한 머신 러닝 모델입니다. DNN은 적은 수의 단계로 임의의 병렬 연산을 수행할 수 있기 때문에 강력합니다. DNN의 강력한 성능을 보여주는 놀라운 예는 2진법 크기의 숨겨진 레이어 2개만 사용하여 N비트 숫자를 정렬하는 능력입니다[27]. 따라서 신경망은 기존의 통계 모델과 관련이 있지만, 복잡한 계산을 학습합니다. 또한 레이블이 지정된 훈련 집합에 네트워크의 매개변수를 지정하기에 충분한 정보가 있을 때마다 대규모 DNN을 감독된 역전파를 통해 훈련할 수 있습니다. 따라서 좋은 결과를 얻을 수 있는 대규모 DNN의 매개변수 설정이 존재하는 경우(예: 사람이 작업을 매우 빠르게 해결할 수 있기 때문), 지도형 역전파는 이러한 매개변수를 찾아 문제를 해결합니다.

유연성과 강력한 성능에도 불구하고 DNN은 입력과 목표가 고정된 차원의 벡터로 현명하게 인코딩될 수 있는 문제에만 적용될 수 있습니다. 많은 중요한 문제는 길이를 미리 알 수

없는 시퀀스로 표현하는 것이 가장 효과적이기 때문에 이는 중요한 한계입니다. 예를 들어 음성 인식과 기계 번역은 순차적 문제입니다. 마찬가지로 질문에 대한 답변도 질문을 나타내는 단어 시퀀스를

답을 나타내는 단어의 시퀀스입니다. 따라서 시퀀스를 시퀀스에 매핑하는 방법을 학습하는 도메인 독립적인 방법이 유용할 것임이 분명합니다.

시퀀스는 입력과 출력의 차원이 알려져 있고 고정되어 있어야 하기 때문에 DNN에 어려운 문제를 제기합니다. 이 논문에서는 장단기 메모리(LSTM) 아키텍처[16]를 간단하게 적용하면 일반적인 시퀀스 간 문제를 해결할 수 있음을 보여줍니다. 이 아이디어는 한 번에 한 타임스텝씩 입력 시퀀스를 읽어 큰 고정 차원 벡터 표현을 얻은 다음, 다른 LSTM을 사용하여 해당 벡터에서 출력 시퀀스를 추출하는 것입니다(그림 1). 두 번째 LSTM은 입력 시퀀스에 따라 조건이 지정된다는 점을 제외하면 본질적으로 순환 신경망 언어 모델입니다[28, 23, 30]. 장거리 시간 종속성이 있는 데이터에 대해 성공적으로 학습할 수 있는 LSTM의 능력은 입력과 해당 출력 사이에 상당한 시간 지연이 있기 때문에 이 애플리케이션에 자연스러운 선택입니다(그림 1).

신경망으로 일반적인 시퀀스 대 시퀀스 학습 문제를 해결하기 위한 많은 관련 시도가 있었습니다. 우리의 접근 방식은 전체 입력 문장을 벡터에 매핑한 최초의 연구자인 Kalchbrenner와 Blunsom[18]과 밀접한 관련이 있으며, 후자는 구문 기반 시스템에서 생성된 가설을 점수화하는 데만 사용되었지만 Cho 등[5]과도 관련이 있습니다. 그레이브스[10]는 신경망이 입력의 서로 다른 부분에 집중할 수 있는 새로운 차별적 주의 메커니즘을 도입했으며, 이 아이디어의 우아한 변형은 바다나우 등[2]에 의해 기계 번역에 성공적으로 적용되었습니다. 연결주의 시퀀스 분류는 신경망을 사용하여 시퀀스를 시퀀스에 매핑하는 또 다른 인기 있는 기법이지만 입력과 출력 간의 단조로운 정렬을 가정합니다[11].

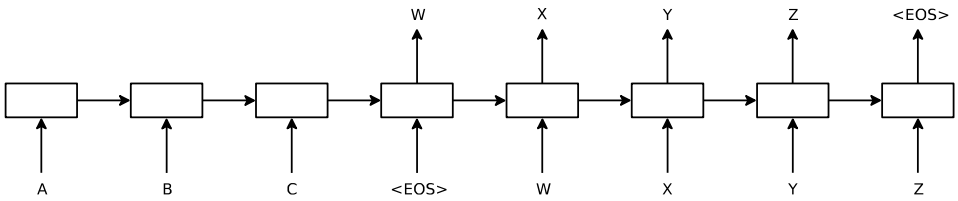

그림 1: 모델은 입력 문장 "ABC"를 읽고 출력 문장으로 "WXYZ"를 생성합니다. 모델은 문장 끝 토큰을 출력한 후 예측을 중단합니다. 입력 문장을 거꾸로 읽으면 데이터에 단기 종속성이 많이 발생하여 최적화 문제가 훨씬 쉬워지기 때문에 LSTM은 입력 문장을 거꾸로 읽습니다.

이 작업의 주요 결과는 다음과 같습니다. WMT'14 영어에서 프랑스어로의 번역 작업에서 간단한 왼쪽에서 오른쪽 빔 검색 디코더를 사용하여 5개의 심층 LSTM 앙상블(각각 384M 파라미터와 8,000 차원 상태)에서 번역을 직접 추출하여 **34.81의** BLEU 점수를 얻었습니다. 이는 대규모 신경망을 사용한 직접 번역으로 얻을 수 있는 최고의 결과입니다. 비교를 위해, 이 데이터 세트에서 SMT 기준선의 BLEU 점수는 33.30입니다[29]. 8만 단어의 어휘를 가진 LSTM이 34.81의 BLEU 점수를 얻었으므로 참조 번역에 이 8만 단어에 포함되지 않은 단어가 포함될 때마다 점수가 감점되었습니다. 이 결과는 개선의 여지가 많은 상대적으로 최적화되지 않은 소어휘 신경망 아키텍처가 구문 기반 SMT 시스템보다 성능이 뛰어나다는 것을 보여줍니다.

마지막으로, 동일한 작업에 대해 공개적으로 사용 가능한 SMT 기준선의 1000개 베스트 리스트의 점수를 다시 매기기 위해 LSTM을 사용했습니다[29]. 이를 통해 36.5의 BLEU 점수를 얻었으며, 이는 기준선을 다음과 같이 개선했습니다.
3.2점이며 이 작업에 대해 이전에 발표된 최고 결과(37.0점[9])에 근접합니다.

놀랍게도 최근 관련 아키텍처를 연구한 다른 연구자들의 경험[26]에 따르면, LSTM은 매우 긴 문장에서 어려움을 겪지 않았습니다. 긴 문장을 잘 처리할 수 있었던 이유는 학습 및 테스트 세트에서 목표 문장이 아닌 원본 문장의 단어 순서를 뒤집었기 때문입니다. 이렇게 함으로써 최적화 문제를 훨씬 더 단순하게 만드는 많은 단기 종속성을 도입했습니다(2절과 3.3절 참조). 그 결과, SGD는 긴 문장에도 문제가 없는 LSTM을 학습할 수 있었습니다. 소스 문장의 단어를 뒤집는 간단한 트릭은 이 작업의 핵심적인 기술적 기여 중 하나입니다.

LSTM의 유용한 특성은 가변 길이의 입력 문장을 고정 차원 벡터 표현으로 매핑하는 방법을 학습한다는 점입니다. 번역이 원본 문장을 의역하는 경향이 있다는 점을 감안할 때, 번역

목표에서는 의미가 비슷한 문장은 서로 가깝지만 다른 문장은 서로 다르기 때문에 LSTM이 그 의미를 포착하는 문장 표현을 찾도록 장려합니다.

문장의 의미는 멀리 떨어져 있을 것입니다. 정성적 평가는 이 주장을 뒷받침하며, 우리 모델이 어순을 인식하고 능동태와 피동태에 상당히 변함이 없음을 보여줍니다.

2 모델

순환신경망(RNN)[31, 28]은 피드포워드 신경망을 시퀀스로 자연스럽게 일반화한 것입니다. 입력 시퀀스($x_1, \ldots, x_T$)가 주어지면 표준 RNN은 다음 방정식을 반복하여 출력 시퀀스($y_1, \ldots, y_T$)를 계산합니다:

$$h_t = \text{시그마}\left(W^{\text{hx}} x_t + W^{\text{hh}} h_{t-1}\right) \qquad y_t = W^{\text{yh}} h_t$$

입력과 출력 사이의 정렬을 미리 알고 있는 경우 RNN은 시퀀스를 시퀀스에 쉽게 매핑할 수 있습니다. 그러나 입력과 출력 시퀀스의 길이가 서로 다르고 복잡하고 단조롭지 않은 관계인 문제에 RNN을 적용하는 방법은 명확하지 않습니다.

일반적인 시퀀스 학습을 위한 가장 간단한 전략은 하나의 RNN을 사용하여 입력 시퀀스를 고정된 크기의 벡터에 매핑한 다음 다른 RNN을 사용하여 벡터를 목표 시퀀스에 매핑하는 것입니다(이 접근 방식은 Cho 등[5]에서도 사용되었습니다). 이 방법은 모든 관련 정보가 RNN에 제공되기 때문에 원칙적으로 작동할 수 있지만, 장기적인 종속성(그림 1)으로 인해 RNN을 훈련하기가 어렵습니다[14, 4, 16, 15]. 그러나 장단기 메모리(LSTM)
[16]은 장거리 시간 종속성 문제를 학습하는 것으로 알려져 있으므로 이 설정에서는 LSTM이 성공할 수 있습니다.

LSTM의 목표는 조건부 확률 $p(y_1, \ldots, y_{T'} | x_1, \ldots, x_T)$를 추정하는 것입니다. 여기서 ($x_1, \ldots, x_T$)는 입력 시퀀스이고 $y_1, \ldots, y_{T'}$는 길이 T'가 T와 다를 수 있는 해당 출력 시퀀스입니다. LSTM은 먼저 입력 시퀀스 ($x_1, \ldots, x_T$)의 마지막 숨겨진 상태에 의해 주어진 입력 시퀀스의 고정 차원 표현 v를 구하여이 조건부 확률을 계산합니다.
LSTM을 사용한 다음 $y_1, \ldots, y'$의 확률을 계산합니다. , $y_{T'}$의 확률을 계산하고 초기 숨겨진 상태가 $x_1, , \ldots, x_T$:

$$p(y_1, \ldots, y_{T'} | x_1, \ldots, x_T) = \prod_{t=1}^{T'} p(y_t | v, y_1, \ldots, y)_{t-1} \tag{1}$$

이 방정식에서 각 $p(y_t | v, y_1, \ldots, y_{t-1})$ 분포는 어휘의 모든 단어에 대해 소프트 맥스로 표현됩니다. 여기서는 Graves [10]의 LSTM 공식을 사용합니다. 각 문장은 특수 문장 끝 기호 "<EOS>"로 끝나야 하며, 이를 통해 모델이 다음을 수행할 수 있습니다.
는 가능한 모든 길이의 시퀀스에 대한 분포를 정의합니다. 전체 체계는 그림 1에 요약되어 있으며, 표시된 LSTM은 "A", "B", "C", "<EOS>"의 표현을 계산한 다음 이 표현을 사용하여 "W", "X", "Y", "Z", "<EOS>"의 확률을 계산합니다.

실제 모델은 위의 설명과 세 가지 중요한 점에서 다릅니다. 첫째, 입력 시퀀스와 출력 시퀀스에 각각 다른 두 개의 LSTM을 사용했는데, 이는 무시할 수 있는 계산 비용으로 모델 파라미터 수를 늘리고 여러 언어 쌍에 대해 동시에 LSTM을 훈련하는 것이 자연스럽기 때문입니다[18]. 둘째, 딥 LSTM이 얕은 LSTM보다 성능이 월등히 뛰어나다는 것을 발견했기 때문에 4계층으로 구성된 LSTM을 선택했습니다. 셋째, 입력 문장의 단어 순서를 뒤집는 것이 매우 유용하다는 것을 발견했습니다. 예를 들어, a, b, c라는 문장을 α, β, γ라는 문장에 매핑하는 대신 α, β, γ는 a, b, c의 번역인 c, b, a를 α, β, γ에 매핑하도록 LSTM에 요청합니다. 이렇게 하면 a는 α에 가깝고 b는 β에 상당히 가깝기 때문에 SGD가 입력과 출력 사이에 "통신"을 쉽게 설정할 수 있게 됩니다. 이러한 간단한 데이터 변환을 통해 LSTM의 성능을 크게 향상시킬 수 있습니다.

3 실험

WMT'14 영어-프랑스어 MT 과제에는 두 가지 방법으로 이 방법을 적용했습니다. 레퍼런스

SMT 시스템을 사용하지 않고 입력 문장을 직접 번역하는 방법과 SMT 기준선의 n-최적 목록을 재채점하는 방법에 사용했습니다. 이러한 번역 방법의 정확도를 보고하고, 샘플 번역을 제시하며, 결과 문장 표현을 시각화합니다.

3.1 데이터 세트 세부 정보

WMT'14 영어-프랑스어 데이터셋을 사용했습니다. 3억 4,800만 개의 프랑스어 단어와 3억 4,000만 개의 영어 단어로 구성된 1,200만 개의 센텐스 하위 집합으로 모델을 학습시켰는데, 이는 [29]에서 깨끗하게 "선택된" 하위 집합입니다. 이 번역 작업과 특정 훈련 세트 하위 집합을 선택한 이유는 기준 SMT[29]의 1000개 베스트 목록과 함께 토큰화된 훈련 및 테스트 세트가 공개적으로 제공되기 때문입니다.

일반적인 신경 언어 모델은 각 단어에 대한 벡터 표현에 의존하기 때문에 두 언어 모두에 고정 어휘를 사용했습니다. 소스 언어에는 가장 빈번한 단어 16만 개를, 타깃 언어에는 가장 빈번한 단어 8만 개를 사용했습니다. 어휘에서 벗어난 모든 단어는 특수한 "UNK" 토큰으로 대체했습니다.

3.2 디코딩 및 재스코어링

실험의 핵심은 많은 문장 쌍에 대해 대규모 심층 LSTM을 훈련하는 것이었습니다. 소스 문장 S가 주어졌을 때 올바른 번역 T의 로그 확률을 최대화하여 훈련했기 때문에 훈련 목표는 다음과 같습니다.

$$1/|S| \sum_{(T,S)\in S} \text{로그}\, p(T|S)$$

여기서 S는 훈련 세트입니다. 학습이 완료되면 LSTM에 따라 가장 가능성이 높은 번역을 찾아 번역을 생성합니다:

$$\hat{T} = \arg\max_{T} p(T|S) \quad (2)$$

여기서 부분 가설은 일부 번역의 접두사이며, 소수의 부분 가설 B를 유지하는 간단한 왼쪽에서 오른쪽 빔 검색 디코더를 사용하여 가장 가능성이 높은 번역을 검색합니다. 각 타임스텝에서 빔의 각 부분 가설을 어휘에서 가능한 모든 단어로 확장합니다. 이렇게 하면 가설의 수가 크게 증가하므로 모델의 로그 확률에 따라 가장 가능성이 높은 가설 B를 제외한 모든 가설을 폐기합니다. 가설에 "<EOS>" 기호가 추가되면 해당 가설은 빔에서 제거되고 완전한 가설 집합에 추가됩니다. 이 디코더는 근사치이지만 구현은 간단합니다. 흥미롭게도 이 시스템은 빔 크기가 1인 경우에도 잘 작동하며, 빔 크기가 2인 경우 빔 검색의 대부분의 이점을 제공합니다(표 1).

또한 기준 시스템[29]에서 생성된 1000개의 베스트 리스트의 점수를 다시 매기는 데에도 LSTM을 사용했습니다. n개의 베스트 리스트에 대한 점수를 다시 매기기 위해 모든 가설의 로그 확률을 LSTM으로 계산한 다음, 해당 가설의 점수와 LSTM의 점수를 짝수 평균으로 구했습니다.

3.3 소스 문장 뒤집기

LSTM은 장기 종속성이 있는 문제를 해결할 수 있지만, 소스 문장이 반전될 때(목표 문장은 반전되지 않을 때) 훨씬 더 잘 학습한다는 사실을 발견했습니다. 이를 통해 LSTM의 테스트 난해도는 5.8에서 4.7로 떨어졌고, 디코딩된 번역의 테스트 BLEU 점수는 25.9에서 30.6으로 상승했습니다.

이 현상에 대한 완전한 설명은 없지만, 데이터 세트에 많은 단기 종속성이 도입되었기 때문이라고 생각합니다. 일반적으로 소스 문장과 목표 문장을 연결할 때 소스 문장의 각 단어는 목표 문장의 해당 단어에서 멀리 떨어져 있습니다. 그 결과, 이 문제는 "최소 시간 지연"이 큽니다[17]. 소스 문장의 단어를 반대로 바꾸면 소스 언어와 목표 언어의 해당 단어 사이의 평균 거리는 변하지 않습니다. 그러나 소스 언어의 처음 몇 단어는 이제 대상 언어의 처음 몇 단어에 매우 가깝기 때문에 문제의 최소 시간 지연이 크게 줄어듭니다. 따라서 역전파를 통해 소스 문장과 대상 문장 간의 '통신 설정'이 더 쉬워져 전반적인 성능이 크게 향상됩니다.

처음에는 입력 문장을 뒤집으면 목표 문장의 앞부분에서만 예측 정확도가 높아지고 뒷부분에서는 예측 정확도가 떨어질 것이라고 생각했습니다. 하지만 반전된 소스 문장으로 훈련된 LSTM은 긴 문장을 훨씬 더 잘 예측했습니다.

을 원시 소스 문장으로 훈련한 결과(3.7절 참조), 입력 문장을 뒤집으면 메모리 활용도가 더 높은 LSTM이 생성된다는 것을 알 수 있습니다.

3.4 교육 세부 정보

LSTM 모델은 상당히 쉽게 훈련할 수 있다는 것을 발견했습니다. 각 레이어에 1000개의 셀과 1000개의 차원 단어 임베딩이 있는 4개의 레이어, 16만 개의 입력 어휘와 8만 개의 출력 어휘가 있는 심층 LSTM을 사용했습니다. 따라서 딥 LSTM은 8000개의 실수를 사용해 문장을 표현합니다. 딥 LSTM이 훨씬 더 큰 숨겨진 상태 때문에 레이어가 추가될 때마다 난해도가 10% 가까이 감소하는 얕은 LSTM보다 훨씬 뛰어난 성능을 보이는 것으로 나타났습니다. 각 출력에 80,000단어 이상의 나이브 소프트맥스를 사용했습니다. 그 결과 LSTM의 파라미터는 384만 개이며, 이 중 64만 개는 순수 반복 연결입니다("인코더" LSTM의 경우 32만 개, "디코더" LSTM의 경우 32만 개). 전체 훈련 세부 사항은 아래에 나와 있습니다:

- 모든 LSTM의 파라미터를 -0.08에서 0.08 사이의 균일한 분포로 초기화했습니다.
- 모멘텀이 없는 확률론적 경사 하강을 사용했으며 학습률은 0.7로 고정했습니다. 5개의 에포크 이후에는 반 에포크마다 학습 속도를 절반으로 줄이기 시작했습니다. 총 7.5개의 에포크 동안 모델을 훈련했습니다.
- 그라데이션에 128개의 시퀀스를 배치로 사용하고 배치의 크기(즉, 128)로 나누었습니다.
- LSTM은 소실 그라디언트 문제를 겪지 않는 경향이 있지만, 폭발적인 그라디언트를 가질 수 있습니다. 따라서 우리는 기울기의 규범이 임계값을 초과하면 스케일링하여 기울기의 규범에 엄격한 제약을 가했습니다[10, 25]. 각 훈련 배치에 대해 다음을 계산합니다.
  $s = \|g\|_2$, 여기서 g는 기울기를 128로 나눈 값입니다. $s > 5$인 경우 $g = \frac{5g}{s}$로 설정합니다.
- 문장마다 길이가 다릅니다. 대부분의 문장은 짧지만(예: 길이 20~30) 일부 문장은 길기 때문에(예: 길이 100 *초과*) 무작위로 선택된 128개의 미니 배치가 있습니다. 훈련 문장에는 짧은 문장이 많고 긴 문장이 적기 때문에 미니배치에서 많은 연산이 낭비됩니다. 이 문제를 해결하기 위해 미니배치에 포함된 모든 문장의 길이가 거의 같도록 하여 속도를 2배로 향상시켰습니다.

3.5 병렬화

이전 섹션에서 설명한 구성으로 딥 LSTM을 C++로 구현하여 싱글 GPU에서 처리하면 초당 약 1,700개의 단어를 처리할 수 있습니다. 이는 우리의 목적에 비해 너무 느리기 때문에 8-GPU 머신을 사용하여 모델을 병렬화했습니다. LSTM의 각 레이어는 서로 다른 GPU에서 실행되었으며, 계산이 완료되는 즉시 다음 GPU/레이어에 활성화 정보를 전달했습니다. 우리 모델에는 4개의 LSTM 레이어가 있으며, 각 레이어는 별도의 GPU에 상주합니다. 나머지 4개의 GPU는 소프트맥스를 병렬화하는 데 사용되었으므로 각 GPU는 1000 × 20000 매트릭스를 곱하는 역할을 담당했습니다. 그 결과 6,300의 속도를 구현했습니다.
(영어와 프랑스어 모두) 단어를 128개의 미니 배치 크기로 초당 처리했습니다. 교육에는 약 10분이 소요되었습니다.
일 동안 구현되었습니다.

3.6 실험 결과

번역의 품질을 평가하기 위해 대문자로 표기된 BLEU 점수[24]를 사용했습니다. *토큰화된* 예측과 지상 실측 데이터에 대해 `multi-bleu.pl`[1] 을 사용하여 BLEU 점수를 계산했습니다. 이 BELU 점수 평가 방식은 [5] 및 [2]와 일치하며 [29]의 33.3점을 재현합니다. 그러나 이러한 방식으로 최고의 WMT'14 시스템[9](`statmt.org\matrix`에서 예측을 다운로드할 수 있음)을 평가하면 37.0을 얻게 되며, 이는 `statmt.org\matrix`에서 보고한 35.8보다 더 높습니다.

결과는 표 1과 2에 나와 있습니다. 무작위 초기화와 미니배치의 무작위 순서가 다른 LSTM 앙상블을 사용하여 최상의 결과를 얻었습니다. LSTM 앙상블의 디코딩된 번역이 최고의 WMT'14 시스템을 능가하지는 못하지만, 대규모 MT에서 순수 신경망 번역 시스템이 구문 기반 SMT 기준선을 능가한 것은 이번이 처음입니다.

[1]BLEU 점수에는 여러 가지 변형이 있으며, 각 변형은 펄 스크립트로 정의됩니다.

방법	BLEU 점수 테스트(ntst14)
바흐다나우 등 [2]	28.45
기준선 시스템 [29]	33.30
단일 포워드 LSTM, 빔 크기 12	26.17
단일 역방향 LSTM, 빔 크기 12	30.59
5개의 역방향 LSTM으로 구성된 앙상블, 빔 크기 1	33.00
2개의 역방향 LSTM으로 구성된 앙상블, 빔 크기 12	33.27
5개의 역방향 LSTM으로 구성된 앙상블, 빔 크기 2	34.50
5개의 역방향 LSTM으로 구성된 앙상블, 빔 크기 12	**34.81**

표 1: WMT'14 영어-프랑스어 테스트 세트(ntst14)에서의 LSTM 성능. 빔 크기가 2인 5개의 LSTM으로 구성된 앙상블이 빔 크기가 12인 단일 LSTM보다 저렴하다는 점에 유의하세요.

방법	BLEU 점수 테스트(ntst14)
기준선 시스템 [29]	33.30
조 외 [5]	34.54
WMT'14 최고 결과 [9]	**37.0**
단일 포워드 LSTM으로 베이스라인 1000-최고점 기록하기	35.61
단일 역 LSTM으로 기준선 1000-최고점 기록하기	35.85
5개의 역방향 LSTM으로 구성된 앙상블로 기준선 1000-최고 점수 얻기	**36.5**
기준 1000대 베스트 리스트의 오라클 재채점	~45

표 2: WMT'14 영어-프랑스어 테스트 세트(ntst14)에서 SMT 시스템과 함께 신경망을 사용하는 방법.

작업을 상당한 차이로 앞섰지만, 어휘를 벗어난 단어는 처리하지 못했습니다. 기준 시스템의 1000개 베스트 목록을 재채점하는 데 사용된 LSTM은 최고 WMT'14 결과의 0.5 BLEU 포인트 이내입니다.

3.7 긴 문장에 대한 성능

그림 3에서 정량적으로 볼 수 있듯이 긴 문장에서도 LSTM이 잘 작동하는 것을 발견하고 놀랐습니다. 표 3은 긴 문장의 몇 가지 예와 그 번역을 보여줍니다.

3.8 모델 분석

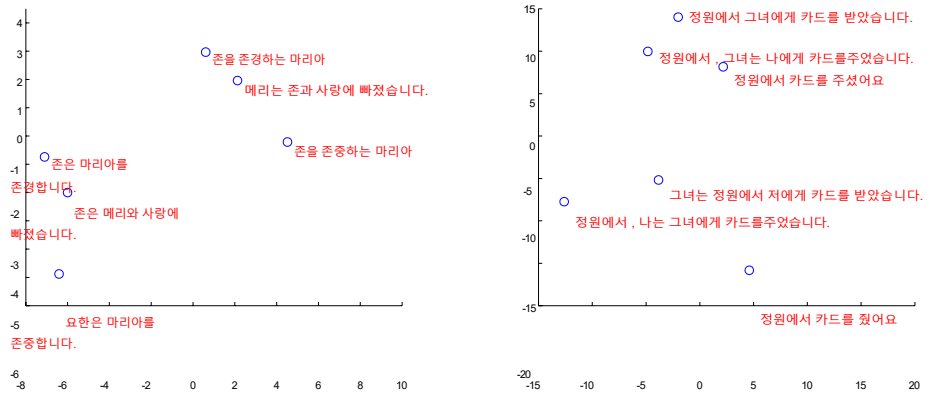

그림 2: 그림은 그림의 구문을 처리한 후 얻은 LSTM 숨겨진 상태의 2차원 PCA 투영을 보여줍니다.

구문은 의미별로 클러스터링되어 있는데, 이 예에서는 주로 어순의 함수이며, 단어 집합 모델로는 포착하기 어렵습니다. 두 클러스터의 내부 구조가 비슷하다는 것을 알 수 있습니다.

이 모델의 매력적인 특징 중 하나는 단어 시퀀스를 고정된 차원의 벡터로 변환할 수 있다는 점입니다. 그림 2는 학습된 표현 중 일부를 시각화한 것입니다. 이 그림은 표현이 단어의 순서에는 민감하지만

유형	문장
모델	자동차 제조업체 Audi의 관리위원회 위원 인 Ulrich UNK는 휴대용 휴대 전화가 거리 측정 장치로 사용되지 않도록 관리위원회 회의 전에 수거 할 수 있도록 몇 년 전부터 실용적인 관행이 필요하다고 주장합니다.
진실	울리히 하켄베르그 , 자동차 제조업체 Audi의 관리 책임자 , de´clare que la collecte des te´le´phones portables avant les re´unions du conseil , afin qu' ils ne puissent pas e^tre utilise´s comme appareils d' e´coute a` distance , est une pratique courante depuis des anne´es .
모델	" 휴대 전화는 그 자체로 질문이 아닙니다. 잠재적으로 내비게이션 장치와 간섭을 일으킬 수 있지만 FCC에 따르면 "우리는 그들이 공기 중에있을 때 휴대 전화와 간섭을 일으킬 수 있음을 알고 있습니다."라고 UNK는 말합니다.
진실	" 휴대용 전화기는 그 자체로 문제가되지 않습니다. pourraient e´ventuellement cre´er des interfe´rences avec les instruments de navigation , mais parce que nous savons , d' apre`s la FCC , qu' ils pourraient perturber les antennes-relais de te´le´phonie mobile s' ils sont utilise´s a` bord " , a de´clare´ Rosenker .
모델	아브르 라 크리에이션 , il ya는 " 폭력 감정 콘트 르 코프 드 유니 에테르 체르 " , 이는 "테이프와 함께 제공되는 "컴포지션 프로세스"를 대신하여 몇 시간 안에 "파일 더미"로 축소"됩니다.
진실	Il ya , avec la cre´ation ,"une violence faite a corps aime´ " , 이는 단 몇 시간 만에 "하나의 문자열로 축소"되고, "하루의 단계를 수반하는" 작곡 프로세스가 열리지 않습니다.

표 3: 지상 실측 번역과 함께 LSTM이 생성한 긴 번역의 몇 가지 예입니다. 독자는 Google 번역을 사용하여 번역이 합리적인지 확인할 수 있습니다.

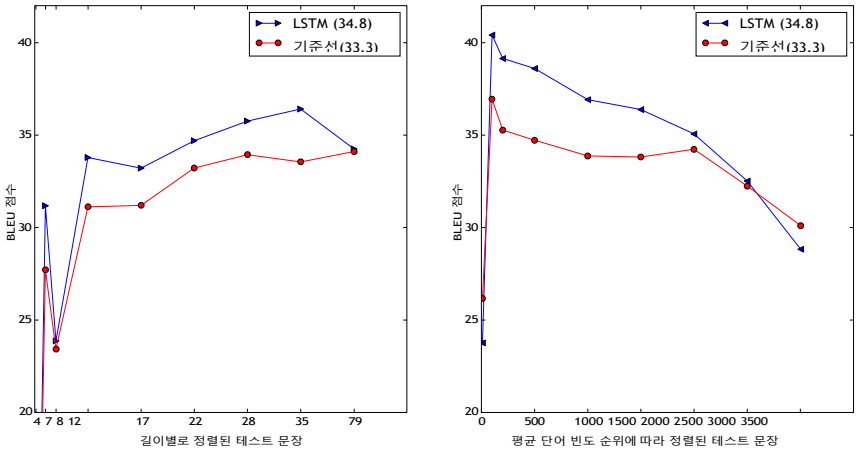

그림 3: 왼쪽 그래프는 문장 길이에 따른 시스템 성능을 보여 주며, X축은 길이별로 정렬된 테스트 문장에 해당하고 실제 시퀀스 길이로 표시되어 있습니다. 35단어 미만의 문장에서는 성능 저하가 없으며, 가장 긴 문장에서는 약간의 성능 저하가 있을 뿐입니다. 오른쪽 그래프는 점점 더 희귀한 단어가 포함된 문장에 대한 LSTM의 성능을 보여주며, 여기서 X축은 '평균 단어 빈도 순위'에 따라 정렬된 테스트 문장에 해당합니다.

능동태를 수동태로 대체합니다. 2차원 투영은 PCA를 사용하여 얻습니다.

4 관련 작업

기계 번역에 신경망을 적용하는 데 대한 많은 연구가 진행 중입니다. 지금까지 가장 간단하고 효과적인 방법은 RNN 언어 모델(RNNLM)[23]을 적용하는 것이거나

MT 작업에 대한 피드포워드 신경망 언어 모델(NNLM)[3]은 강력한 MT 기준선[22]의 n-최적 목록을 점수화하여 번역 품질을 안정적으로 향상시킵니다.

최근에는 연구자들이 소스 언어에 대한 정보를 NNLM에 포함시키는 방법을 연구하기 시작했습니다. 이러한 작업의 예로는 입력 문장의 토픽 모델에 NNLM을 결합하여 점수화 성능을 개선한 Auli 등[1]이 있습니다. Devlin 등[8]도 비슷한 접근 방식을 따랐지만, MT 시스템의 디코더에 NNLM을 통합하고 디코더의 정렬 정보를 사용하여 입력 문장에서 가장 유용한 단어를 NNLM에 제공했습니다. 이 접근 방식은 매우 성공적이었으며 기준선에 비해 큰 개선을 이루었습니다.

우리의 작업은 입력 문장을 벡터로 매핑한 다음 다시 문장으로 매핑한 최초의 연구자인 Kalchbrenner와 Blunsom[18]과 밀접한 관련이 있지만, 이들은 단어의 순서를 잃는 컨볼루션 신경망을 사용하여 문장을 벡터에 매핑합니다. 이 연구와 유사하게 Cho 등[5]은 신경망을 SMT 시스템에 통합하는 데 주안점을 두었지만, 문장을 벡터로 매핑하고 다시 문장으로 매핑하는 데 LSTM과 유사한 RNN 아키텍처를 사용했습니다. Bahdanau 등[2]도 조 등[5]이 경험한 긴 문장의 성능 저하를 극복하기 위해 주의 메커니즘을 사용하는 신경망으로 직접 번역을 시도하여 고무적인 결과를 얻었습니다. 마찬가지로 Pouget-Abadie 등[26]은 구문 기반 접근 방식과 유사한 방식으로 원본 문장의 일부를 매끄럽게 번역하는 방식으로 Cho 등[5]의 메모리 문제를 해결하려고 시도했습니다. 우리는 그들이 단순히 거꾸로 된 원문 문장으로 네트워크를 훈련시킴으로써 비슷한 개선을 달성할 수 있을 것으로 생각합니다.

엔드투엔드 학습은 입력과 출력을 피드포워드 네트워크로 표현하고 이를 공간의 유사한 지점에 매핑하는 모델을 사용하는 Hermann 등[12]의 연구에도 초점을 맞추고 있습니다. 그러나 이러한 접근 방식은 번역을 직접 생성할 수 없습니다. 번역을 얻으려면 미리 계산된 문장 데이터베이스에서 가장 가까운 벡터를 찾거나 문장의 점수를 다시 매겨야 합니다.

5 결론

이 연구에서는 어휘가 제한적이고 문제 구조에 대한 가정이 거의 없는 대규모 심층 LSTM이 대규모 MT 작업에서 어휘가 무제한인 표준 SMT 기반 시스템보다 성능이 뛰어나다는 것을 보여주었습니다. MT에 대한 간단한 LSTM 기반 접근 방식의 성공은 충분한 훈련 데이터만 있다면 다른 많은 시퀀스 학습 문제에서도 잘 작동할 수 있다는 것을 시사합니다.

우리는 소스 문장의 단어를 뒤집어서 얻은 개선의 정도에 놀랐습니다. 단기 종속성이 가장 많은 문제 인코딩을 찾는 것이 학습 문제를 훨씬 더 단순하게 만들 수 있다는 결론을 내렸습니다. 특히, 반전되지 않은 번역 문제(그림 1 참조)에 대해서는 표준 RNN을 훈련할 수 없었지만(실험적으로 검증하지는 않았지만), 소스 문장이 반전된 경우에는 표준 RNN을 쉽게 훈련할 수 있을 것으로 예상합니다.

또한 매우 긴 문장을 정확하게 번역하는 LSTM의 능력에 놀랐습니다. 처음에는 제한된 메모리 때문에 긴 문장을 번역하는 데 실패할 것이라고 확신했고, 다른 연구자들도 우리와 유사한 모델로 긴 문장을 번역할 때 성능이 좋지 않다고 보고했습니다[5, 2, 26]. 하지만 반전된 데이터셋으로 훈련된 LSTM은 긴 문장을 번역하는 데 거의 어려움이 없었습니다.

가장 중요한 것은 간단하고 직관적이며 상대적으로 최적화되지 않은 접근 방식이 SMT 시스템을 능가할 수 있다는 것을 보여 주었기 때문에 추가 작업을 통해 번역 정확도를 더욱 높일 수 있을 것입니다. 이러한 결과는 우리의 접근 방식이 다른 까다로운 시퀀스 대 시퀀스 문제에서도 잘 작동할 수 있음을 시사합니다.

6 감사

유용한 의견과 토론을 제공해 주신 새미 벤지오, 제프 딘, 마티유 데빈, 제프리 힌튼, 날 칼치브레너, 탕 루옹, 울프강 마체레이, 라잣 몽가, 빈센트 반호크, 펭 쉬, 보이치치 자렘바, 그리고 Google 브레인 팀에

감사드립니다.

참조

[1] M. Auli, M. Galley, C. Quirk, 및 G. Zweig. 순환 신경망을 사용한 공동 언어 및 번역 모델링. In *EMNLP*, 2013.

[2] D. Bahdanau, K. Cho, Y. Bengio. 정렬 및 번역을 공동으로 학습하는 신경망 기계 번역. *arXiv 사전 인쇄물 arXiv:1409.0473*, 2014.

[3] Y. 벤지오, R. 뒤샤르메, P. 빈센트, C. 조뱅. 신경 확률론적 언어 모델. *기계 학습 연구 저널*, 1137-1155 페이지, 2003.

[4] Y. 벤지오, P. 시마드, P. 프라스코니. 경사 하강으로 장기 종속성을 학습하는 것은 어렵습니다. *IEEE 신경망 트랜잭션*, 5(2):157-166, 1994.

[5] K. Cho, B. Merrienboer, C. Gulcehre, F. Bougares, H. Schwenk, and Y. Bengio. 통계적 기계 번역을 위해 RNN 인코더-디코더를 사용한 구문 표현 학습. *아카이브 프리프린트 arXiv:1406.1078*, 2014.

[6] D. Ciresan, U. Meier, 및 J. Schmidhuber. 이미지 분류를 위한 다중 열 심층 신경망. In *CVPR*, 2012.

[7] G. E. Dahl, D. Yu, L. Deng, 및 A. Acero. 대규모 어휘 음성 인식을 위한 컨텍스트 의존적 사전 학습 심층 신경망. *IEEE 트랜잭션 오디오, 음성 및 언어 처리 - 음성 및 언어 처리를 위한 딥 러닝 특별호*, 2012.

[8] J. Devlin, R. Zbib, Z. Huang, T. Lamar, R. Schwartz, 및 J. Makhoul. 통계적 기계 번역을 위한 빠르고 강력한 신경망 공동 모델. In *ACL*, 2014.

[9] 나디르 두라니, 배리 해도우, 필립 코엔, 케네스 히필드. WMT-14를 위한 에딘버러의 구문 기반 기계 번역 시스템. *WMT*, 2014.

[10] A. 그레이브스. 순환 신경망으로 시퀀스 생성. *아카이브 프리프린트 arXiv:1308.0850*, 2013.

[11] A. 그레이브스, S. 페르난데스, F. 고메즈, 및 J. 슈미드후버. 연결주의 시간 분류: 반복 신경망으로 분할되지 않은 시퀀스 데이터에 라벨링하기. In *ICML*, 2006.

[12] K. M. 헤르만과 P. 블런섬. 단어 정렬이 없는 다국어 분산 표현. In *ICLR*, 2014.

[13] G. 힌튼, L. 덩, D. 유, G. 달, A. 모하메드, N. 자이틀리, A. 시니어, V. 반호크, P. 응우옌, T. Sainath, 및 B. Kingsbury. 음성 인식에서 음향 모델링을 위한 심층 신경망. *IEEE 신호 처리 매거진*, 2012.

[14] S. Hochreiter. 동적 신경망에 대한 연구. *석사 학위 논문, 뮌헨 테크니컬 대학교 모피 정보학 연구소,* 1991.

[15] S. Hochreiter, Y. Bengio, P. Frasconi, and J. Schmidhuber. 순환망의 경사 흐름: 장기 종속성 학습의 어려움, 2001.

[16] S. 호크라이터와 J. 슈미트후버. 장단기 기억. *신경 계산*, 1997.

[17] S. 호크라이터와 J. 슈미드후버. LSTM은 어려운 긴 시간 지연 문제를 해결할 수 있습니다. 1997.

[18] N. 칼치브레너와 P. 블런섬. 반복적 연속 번역 모델. In *EMNLP*, 2013.

[19] A. 크리제프스키, I. 수츠케버, G. E. 힌튼. 심층 컨볼루션 신경망을 이용한 이미지넷 분류. In *NIPS*, 2012.

[20] Q.V. Le, M.A. Ranzato, R. Monga, M. Devin, K. Chen, G.S. Corrado, J. Dean, 및 A.Y. Ng. 대규모 비지도 학습을 사용하여 높은 수준의 특징 구축. In *ICML*, 2012.

[21] Y. 르쿤, L. 보투, Y. 벤지오, 및 P. 하프너. 문서 인식에 적용된 그라디언트 기반 학습. *IEEE 논문집*, 1998.

[22] T. 미콜로프. *신경망에 기반한 통계적 언어 모델.* 박사 학위 논문, 브르노 공과대학교, 2012.

[23] T. Mikolov, M. 카라피아트, L. 버겟, J. 체르노키`, S. 쿠단푸르. 순환 신경망 기반 언어 모델. In *INTERSPEECH*, 1045-1048, 2010.

[24] K. Papineni, S. Roukos, T. Ward, 및 W. J. Zhu. BLEU: 기계 번역의 자동 평가를 위한 방법. In *ACL*, 2002.

[25] R. 파스카누, T. 미콜로프, Y. 벤지오. 순환 신경망 훈련의 어려움에 대해. *arXiv 사전 인쇄물 arXiv:1211.5063*, 2012.

[26] J. Pouget-Abadie, D. Bahdanau, B. van Merrienboer, K. Cho, and Y. Bengio. 자동 분할을 이용한 신경 기계 번역에서 문장 길이의 저주 극복. *arXiv 사전 인쇄물 arXiv:1409.1257*, 2014.

[27] A. 라즈보로프. 작은 깊이 임계값 회로에서. *제3회 스칸디나비아 알고리즘 이론 워크숍*, 1992.

[28] D. 럼멜하트, G. E. 힌튼, 및 R. J. 윌리엄스. 오류를 역전파하여 표현 학습하기. *Nature*, 323(6088):533-536, 1986.

[29] H. Schwenk. 대학 르망. `http://www-lium.univ-lemans.fr/~schwenk/cslm_joint_paper/`, 2014. [온라인; 2014 년 9 월 03 일 액세스].

[30] M. 순더마이어, R. 슐루터, 및 H. 네이. 언어 모델링을 위한 LSTM 신경망. In *INTER- SPEECH*, 2010.

[31] P. 베르보스. 시간을 통한 역전파: 그것이 하는 일과 방법. *Proceedings of IEEE*, 1990.