2025.01.10(금)

NEURAL MACHINE TRANSLATION BY JOINTLY LEARNING TO ALIGN AND TRANSLATE

지능형 소프트웨어 융합 & AI 연구소

게재년도: 2015
게재사이트: ICLR
논문저자
- Dzmitry Bahdanau
  - Jacobs University Bremen, Germany
-KyungHyun Cho, Yoshua Bengio*
   -Universite´ de Montre´al

컴퓨터공학과 201921087 홍승현

ABSTRACT

- Neural Machine Translation(NMT)는 단일 신경망을 사용해 번역 성능을 최적화하는 최근 제안된 접근법임
- 기존의 통계적 기계 번역(SMT)과 달리, NMT는 인코더-디코더 구조를 통해 소스 문장을 고정 길이 벡터로 인코딩한 뒤 디코딩함
- 고정 길이 벡터 사용이 성능 향상의 병목임을 지적하며, 관련 부분을 자동으로 소프트 검색하는 방식을 제안함
- 이를 통해 영어-프랑스어 번역에서 SOTA 시스템과 유사한 성능을 달성함

1. Introduction

- Neural Machine Translation (NMT)의 등장
  - NMT는 Kalchbrenner와 Blunsom(2013), Sutskever et al.(2014), 그리고 Cho et al.(2014b)에 의해 제안된 번역 방식
  - 기존의 구문 기반 번역 시스템(phrase-based translation system)과 달리, NMT는 단일 대규모 신경망을 통해 번역 시스템을 구축함
- 기존 번역 시스템의 한계
  - 구문 기반 번역 시스템은 독립적으로 최적화된 여러 하위 구성 요소로 이루어져 있음
  - 이와 비교해, NMT는 단일 신경망으로 소스 문장을 읽고 적절한 번역을 출력하도록 학습됨
- Encoder–Decoder 구조의 특징
  - 대부분의 NMT 모델은 인코더-디코더 구조임
    - 인코더: 소스 문장을 고정 길이 벡터로 인코딩함
    - 디코더: 고정 길이 벡터를 기반으로 타겟 문장을 생성
  - 전체 시스템은 소스 문장이 주어졌을 때 정확한 번역이 생성될 확률을 최대화하도록 학습됨
- Encoder–Decoder 구조의 문제점
  - 고정 길이 벡터로 소스 문장의 모든 정보를 압축해야 한다는 제약이 존재
  - 특히, 긴 문장의 경우 학습 데이터에 포함된 문장 길이를 초과하면 성능이 크게 저하됨
- 제안된 접근 방식
  - 이 논문은 인코더-디코더 구조의 고정 길이 벡터 문제를 해결하기 위한 확장 모델을 제안
  - 제안된 모델은 번역 과정에서 타겟 단어를 생성할 때 소스 문장에서 관련성이 높은 부분을 자동으로 검색(soft-search)함
  - 이를 통해 모델이 명시적으로 세그먼트를 형성하지 않고도 필요한 정보를 효과적으로 활용할 수 있음
- 연구의 주요 성과
  - 제안된 모델은 기존 구문 기반 번역 시스템에 필적하는 성능을 영어-프랑스어 번역 작업에서 달성함
  - 정성적 분석에서는 모델이 찾아낸 소프트 정렬 결과가 직관적으로 타당한 것으로 나타남

2. BACKGROUND: NEURAL MACHINE TRANSLATION

- 1. 번역의 확률적 정의
  - 번역은 소스 문장이 주어졌을 때, 조건부 확률 P(y|x)를 최대화하는 타겟 문장 y를 찾는 문제로 정의됨
  - 신경 기계 번역(NMT)은 이 조건부 확률을 병렬 데이터로 학습하는 모델을 설계하는 데 중점을 둠
  - 학습된 조건부 확률을 기반으로, 모델은 가장 높은 확률의 타겟 문장을 생성함
- 2. 기존 신경망 기반 접근법
  - 최근 연구들은 신경망을 사용하여 조건부 확률을 직접 학습하는 접근법을 제안했는데 일반적으로 두 가지 주요 구성 요소를 포함함
  - 인코더(Encoder): 소스 문장을 인코딩하여 고정 길이 벡터로 변환
  - 디코더(Decoder): 인코딩된 벡터를 바탕으로 타겟 문장을 디코딩

2. BACKGROUND: NEURAL MACHINE TRANSLATION

- 3. Encoder-Decoder 구조
  - 인코더는 소스 문장을 벡터 c로 요약합니다.
    - RNN을 사용하여 벡터 c는 숨겨진 상태 ht를 기반으로 생성됩니다.
    - hidden state는

$$h_t = f\left(x_t, h_{t-1}\right)$$

    - 로 정의

$$c = q\left(\left\{h_1, \cdots, h_{T_x}\right\}\right),$$

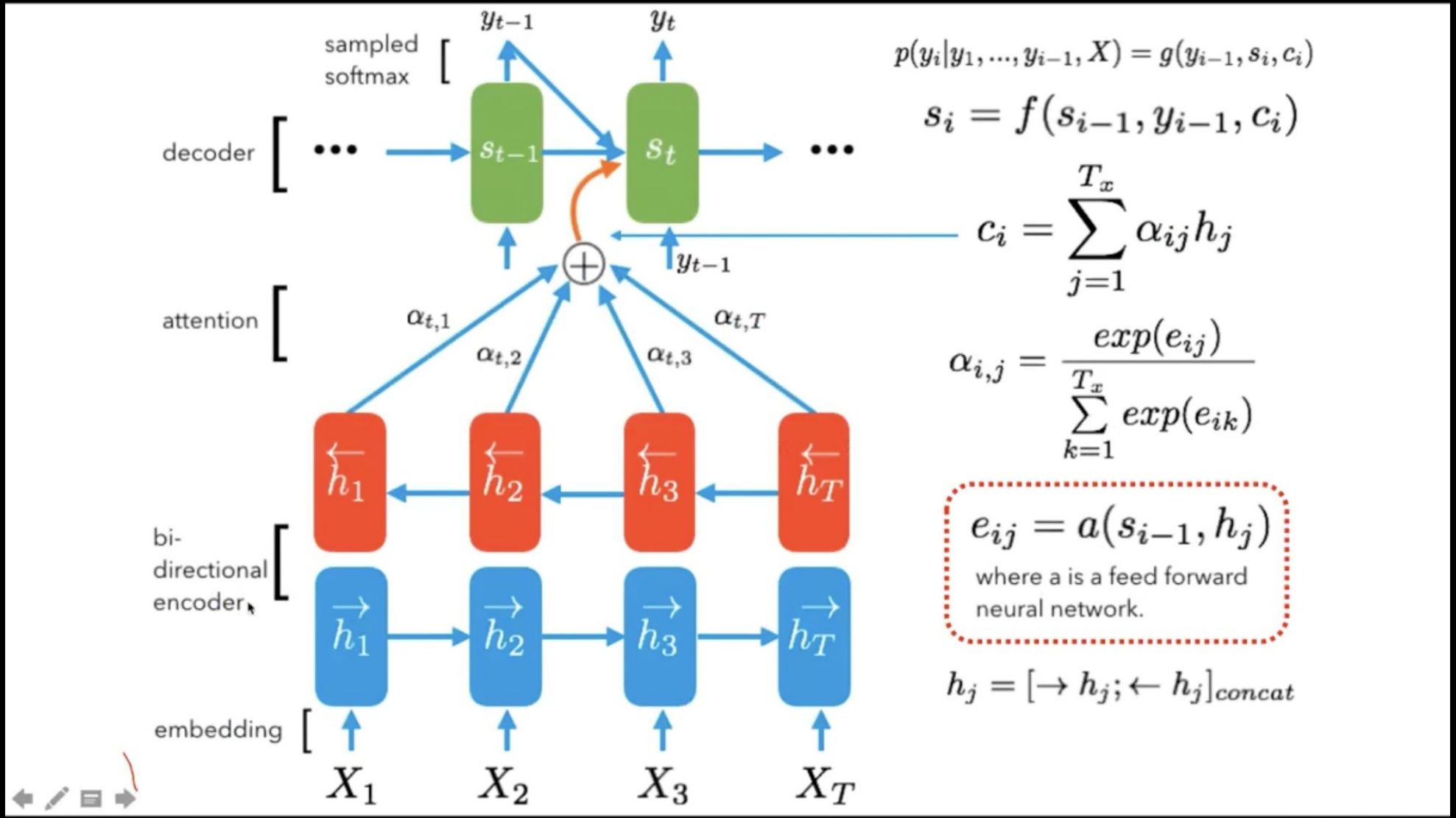
    - 여기서 f, q는 nonlinear functions
  - 디코더는 벡터 c와 이전 타겟 단어를 사용하여 다음 타겟 단어를 예측합니다.
    - 디코더는 조건부 확률을 다음과 같이 분해

$$p(\mathbf{y}) = \prod_{t=1}^{T} p(y_t \mid \{y_1, \cdots, y_{t-1}\}, c),$$

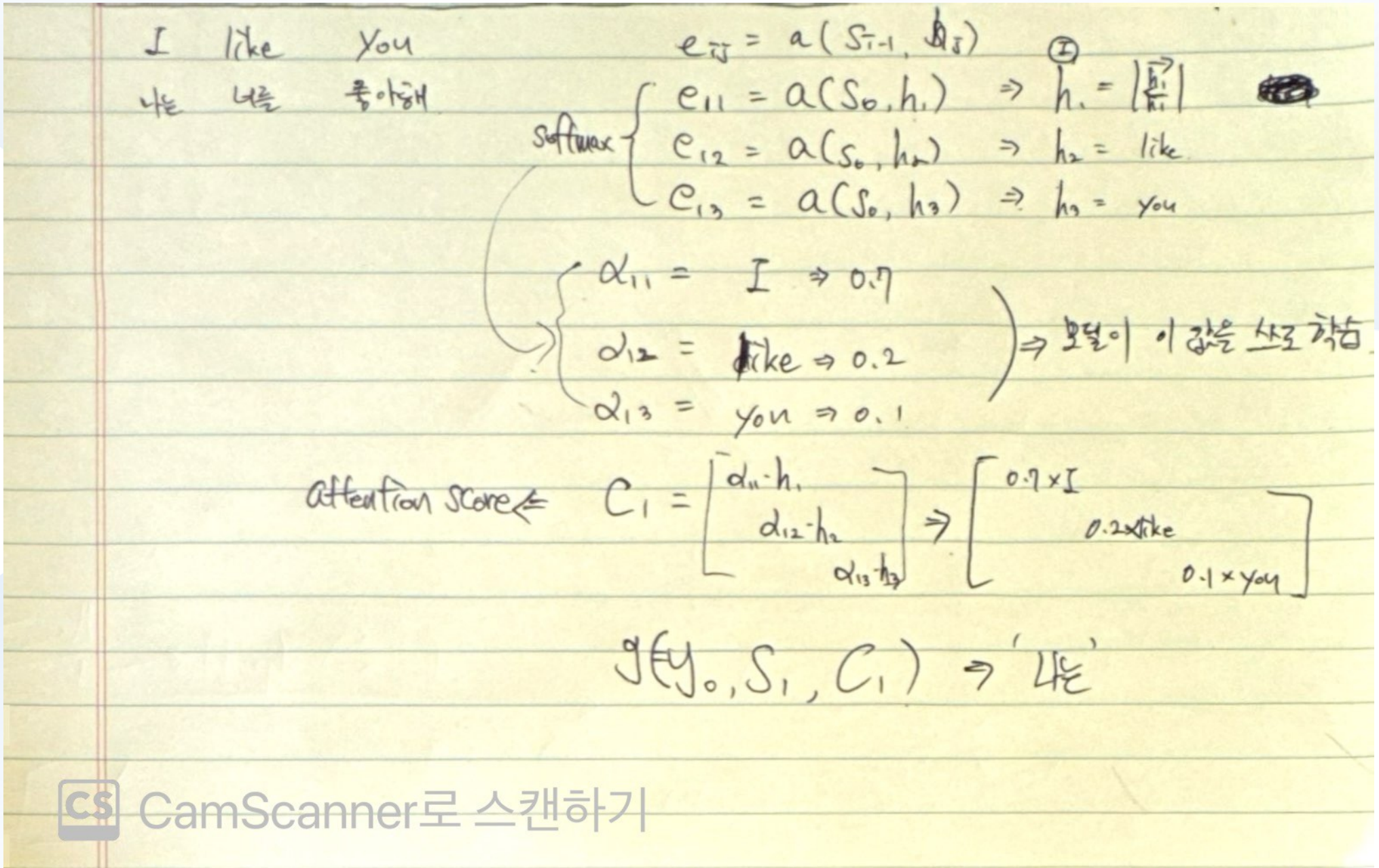
2. BACKGROUND: NEURAL MACHINE TRANSLATION

- 4. Encoder-Decoder 구조의 한계
  - 고정 길이 벡터 c에 소스 문장의 모든 정보를 압축해야 한다는 제약이 있음
  - 특히 긴 문장에서는 벡터 c에 정보를 충분히 담을 수 없기 때문에 번역 품질이 저하
- 5. 기존 한계를 극복하려는 시도
  - 기존 연구에서는 벡터 c를 가변 길이로 사용하는 방법도 가능성을 제시했으나, 이는 명확히 구현되지 않음
  - 본 논문은 이러한 한계를 해결하기 위해 소스 문장을 단일 고정 길이 벡터로 표현하는 대신, 정렬(Alignment)과 번역을 동시에 학습하는 방식을 제안

3. LEARNING TO ALIGN AND TRANSLATE



3. LEARNING TO ALIGN AND TRANSLATE



4. Experiment Settings

- 데이터 출처
  - WMT ’14 영어-프랑스어 병렬 코퍼스 사용
    - Europarl (61M 단어)
    - News commentary (5.5M 단어)
    - UN (421M 단어)
    - 기타 크롤링된 데이터 (총 362.5M 단어)
  - 모든 데이터 합산 시 총 850M 단어로 구성
- 데이터 크기 축소
  - Axelrod et al. (2011)의 데이터 선택 기법을 활용하여 병렬 코퍼스를 348M 단어로 축소
  - 단어 빈도를 기준으로 상위 30,000개의 단어를 선정하여 학습에 사용
  - 선정되지 않은 단어는 [UNK] 토큰으로 대체
- 데이터 분할
  - 학습 데이터: 위에서 선택된 348M 단어의 병렬 코퍼스를 사용
  - 검증 데이터: WMT ’14의 news-test-2012와 news-test-2013을 병합하여 검증 세트로 사용
  - 테스트 데이터: news-test-2014(3003 문장)를 사용하여 모델 성능을 평가
- 전처리
  - 토큰화: 오픈소스 번역 도구인 Moses의 토큰화 스크립트 사용
  - 추가 처리: 소문자 변환, 어간 추출 등의 추가 전처리는 적용하지 않음

4. Experiment Settings

- 비교 모델
  - RNN Encoder–Decoder (RNNencdec)
    - 기존 인코더-디코더 모델
    - 소스 문장을 고정 길이 벡터로 변환 후 디코딩
  - RNNsearch
    - 본 논문에서 제안한 소프트-정렬 기반의 새로운 모델
- 모델 변형
  - 각 모델을 두 가지 문장 길이로 학습
    - RNNencdec-30, RNNsearch-30: 최대 30단어 길이의 문장을 사용하여 학습
    - RNNencdec-50, RNNsearch-50: 최대 50단어 길이의 문장을 사용하여 학습
- 모델 설계
  - RNNencdec:
    - 인코더와 디코더 각각에 1,000개의 히든 유닛 사용
  - RNNsearch:
    - 양방향 인코더(BiRNN): 순방향과 역방향 각각 1,000개의 히든 유닛
    - 디코더: 1,000개의 히든 유닛
    - 타겟 단어 조건부 확률 계산: 단일 maxout 레이어를 포함하는 다층 네트워크 사용

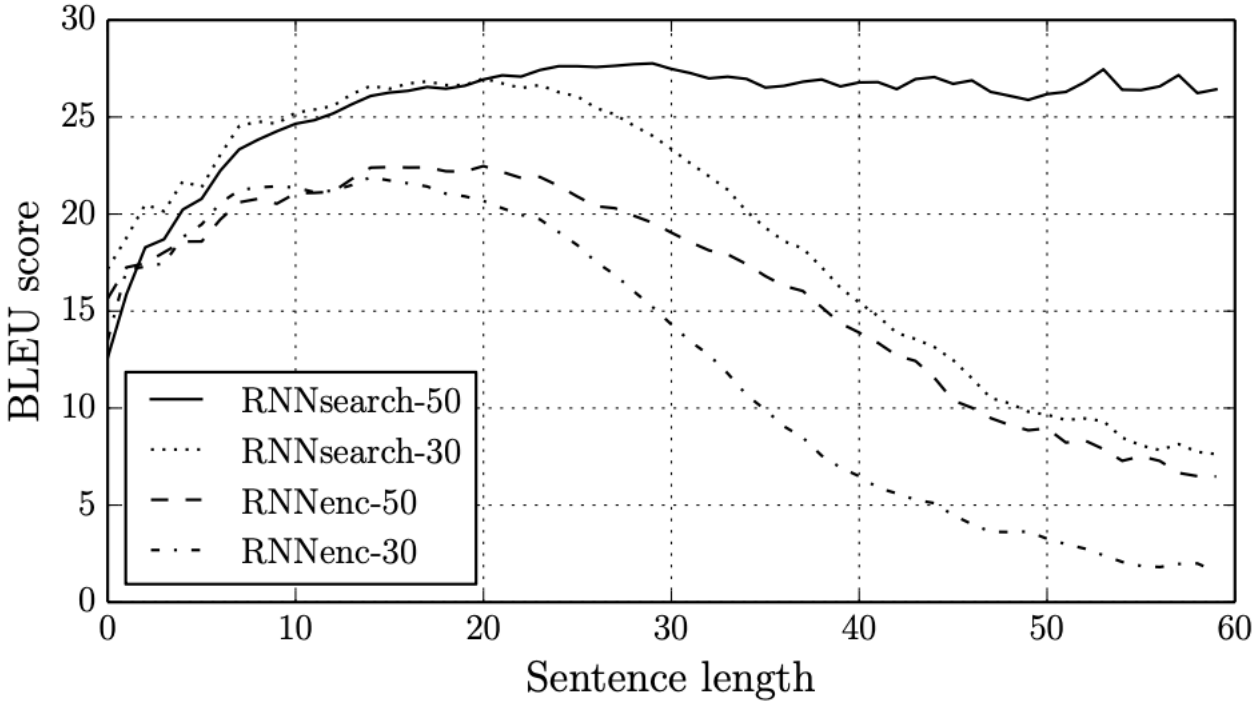
4. Experiment Settings

- 학습 설정
  - 훈련 알고리즘
    - 미니배치 확률적 경사 하강법(SGD) 사용
    - Adadelta(Zeller, 2012) 알고리즘으로 학습 속도를 적응적으로 조정
    - 미니배치 크기: 80개 문장
  - 훈련 시간
    - 각 모델은 약 5일간 학습
    - 학습 데이터의 긴 문장을 포함하기 위해 최대 문장 길이를 기준으로 미니배치를 정렬하여 효율성을 높임
  - 탐색 알고리즘
    - 학습된 모델은 빔 서치(beam search)를 사용하여 번역 생성
    - 빔 서치로 조건부 확률이 최대화되는 번역을 탐색

5. RESULT

Model	All	No UNK°
RNNencdec-30	13.93	24.19
RNNsearch-30	21.50	31.44
RNNencdec-50	17.82	26.71
RNNsearch-50	26.75	34.16
RNNsearch-50⋆	28.45	36.15
Moses	33.30	35.63

Table 1: BLEU scores of the trained models computed on the test set. The second and third columns show respectively the scores on all the sentences and, on the sentences without any unknown word in themselves and in the reference translations. Note that RNNsearch-50⋆ was trained much longer until the performance on the development set stopped improving. (◦) We disallowed the models to generate [UNK] tokens when only the sentences having no unknown words were evaluated (last column).



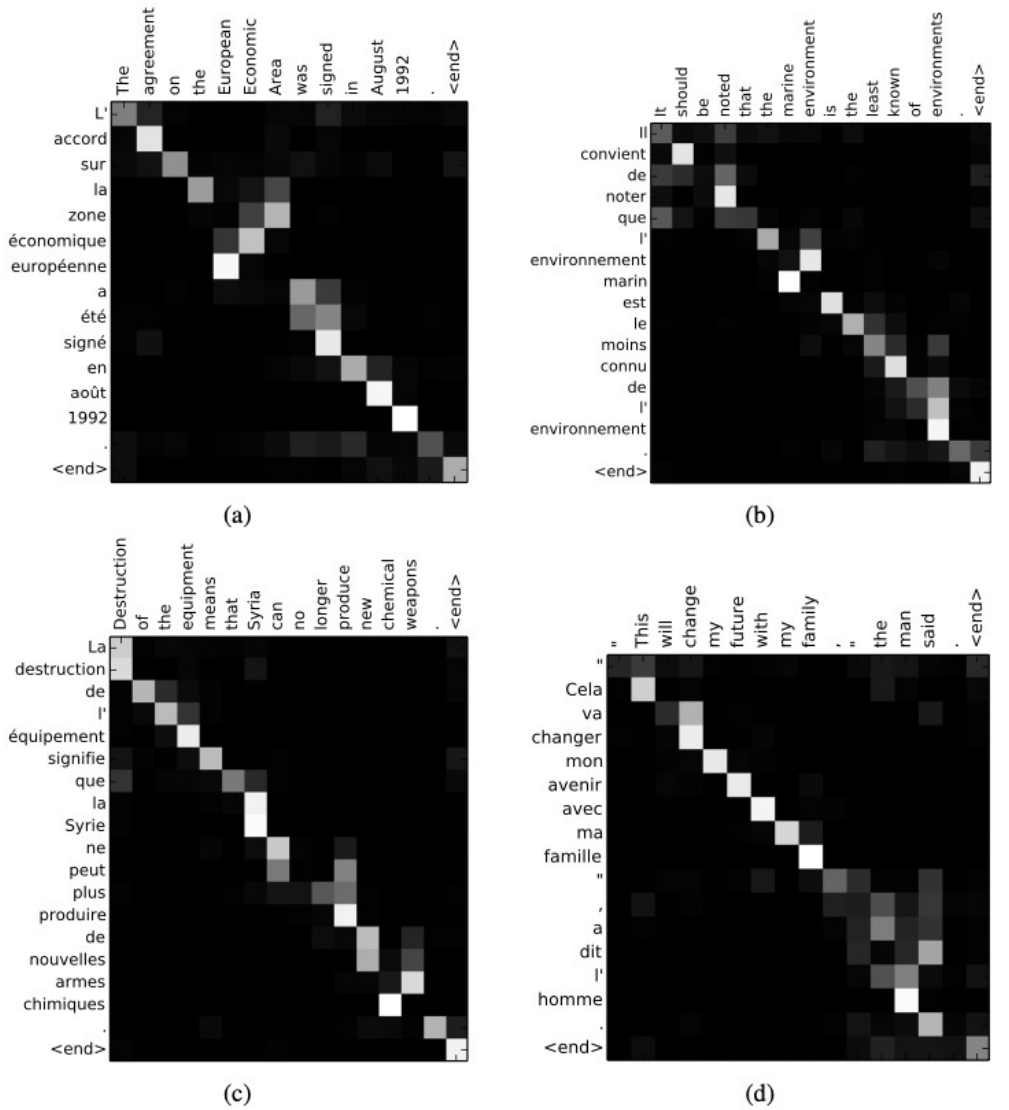


Figure 3: Four sample alignments found by RNNsearch-50. The x-axis and y-axis of each plot correspond to the words in the source sentence (English) and the generated translation (French), respectively. Each pixel shows the weight α_{ij} of the annotation of the j-th source word for the i-th target word (see Eq. (6)), in grayscale (0: black, 1: white). (a) an arbitrary sentence. (b–d) three randomly selected samples among the sentences without any unknown words and of length between 10 and 20 words from the test set.

6. Related Work

- Learning to Align
  - 정렬(alignment)을 학습하는 접근법은 최근 다양한 맥락에서 연구됨
  - Graves(2013)
    - 손글씨 생성(handwriting synthesis) 작업에서 정렬 모델을 도입
    - 입력 문자 시퀀스와 출력 손글씨 간의 정렬을 Gaussian 커널을 사용해 모델링
    - 단, 이 정렬 모델은 **단방향성(monotonicity)**을 가정하여 정렬이 입력 시퀀스의 순서를 항상 유지
    - 제한점: 단방향성은 긴 문장에서 필요한 재배치(reordering)가 요구되는 작업(예: 영어-독일어 번역)에서는 적합하지 않음
  - 본 연구와의 차이점
    - 본 연구의 정렬 모델은 모든 소스 단어에 대해 정렬 점수를 계산
    - 단방향성 가정을 두지 않아, 비모노톤(non-monotonic) 정렬 및 단어 재배치가 가능
    - 소스와 타겟 문장 길이가 다르더라도 적절히 정렬을 수행

6. Related Work

- Neural Networks for Machine Translation
  - 신경망은 과거에 주로 기존 번역 시스템의 부가적 요소로 사용됨
    - 예: 언어 모델로 활용하여 번역 후보군을 재순위화하거나, 구문 점수 계산에 기여
- Schwenk(2012)
  - 신경망을 이용한 구문 점수 계산
    - 신경망이 소스-타겟 구문 쌍의 점수를 계산하고, 이를 기존 구문 기반 번역 시스템의 추가 특징(feature)으로 사용
    - 이는 기존 통계적 번역 시스템에 성능 개선을 가져옴
- Kalchbrenner and Blunsom(2013), Devlin et al.(2014)
  - 기존 번역 시스템의 하위 구성 요소로 신경망을 통합하여 성능을 향상
  - 예: RNN을 사용해 번역 후보군을 재순위화하거나 조건부 확률을 계산
- 전통적인 접근법
  - 신경망은 주로 타겟 측 언어 모델로 사용
    - 후보 번역 리스트를 점수화하거나 재순위화하는 데 기여
    - 예: Schwenk et al.(2006)의 연구

7. Conclusion

- 기존 인코더-디코더 모델의 고정 길이 벡터 문제는 긴 문장에서 번역 품질을 저하
- 이를 해결하기 위해 소프트-검색 기반 정렬을 도입한 RNNsearch 모델을 제안함
- 제안된 모델은 긴 문장에서도 정보 손실을 줄이며 뛰어난 번역 품질을 보여줌
- 영어-프랑스어 번역 작업에서 구문 기반 시스템(Moses)에 필적하는 성능을 달성
- 향후 과제
  - 희귀 단어 처리를 개선
  - NMT의 성능을 다른 언어와 문맥으로 확장

감사합니다!