ELMo(Deep contextualized word representations)

김호재

2024/03/19

TRAIN AND TEST

기존의 Word Embeddings

- 기존에는 각 단어 별로 임베딩을 진행
- 즉, 한 단어에 대해 한가지 뜻밖에 알 수 없었다

	Source	Nearest Neighbors
GloVe	play	playing, game, games, played, players, plays, player, Play, football, multiplayer
biLM	Chico Ruiz made a spectacular play on Alusik 's grounder {...}	Kieffer , the only junior in the group , was commended for his ability to hit in the clutch , as well as his all-round excellent play .
	Olivia De Havilland signed to do a Broadway play for Garson {...}	{...} they were actors who had been handed fat roles in a successful play , and had talent enough to fill the roles competently , with nice understatement .

TRAIN AND TEST

기존의 Word Embeddings

- 이에 따라서 본 논문에서는 2가지 문제를 해결하고자 함.
  - Complex Characteristics (보편적 언어 특성): Syntax&semantics
  - How these uses vary across linguistic contexts(언어적 맥락)

  - 즉, 문맥에 따라 같은 단어더라도 다르게 임베딩을 줄 수 있어야한다.

TRAIN AND TEST

해결 방법

- BiLM이라는 딥러닝 모델을 사용하여 학습
  - 이때, input을 문장단위로 넣게 되어, 문장 단위로 임베딩할 수 있게 함
- 각 입력 단어들의 Vector에 대해 Linear Combination을 학습
  - 즉, 맥락적 의미를 해석하게 함
  - 쉽게 설명하면 각 단어들의 서로에 대한 중요도를 기반의 표현 학습이 이루어짐
- 추가적으로 이전 연구에서 밝혀진 LSTM의 특성을 기반으로 학습 Weight를 정했다.
  - 낮은 단계의 Layer에서는 품사와 같은 문법정보를, 높은 단계의 Layer의 경우 문맥과 관련된 정보를 학습하려는 경향이 있다.

개념설명

- LSTM: 간단히 말해 RNN의 그래디언트 소실 문제를 위해 고안된 모델, 이전 정보를 오랫동안 기억할 수 있는 메모리 셀이 있으며, 이를 통해 긴 시퀀스 처리가 가능하다.
- 현재 토큰을 기준으로 다음 토큰이 나올 확률을 예측하는 모델을 LSTM의 forward부분이라고 함
- 마찬가지로 현재 토큰을 기준으로 이전 토큰을 예측하는 모델을 LSTM의 backward부분이라고 함

Model 설명

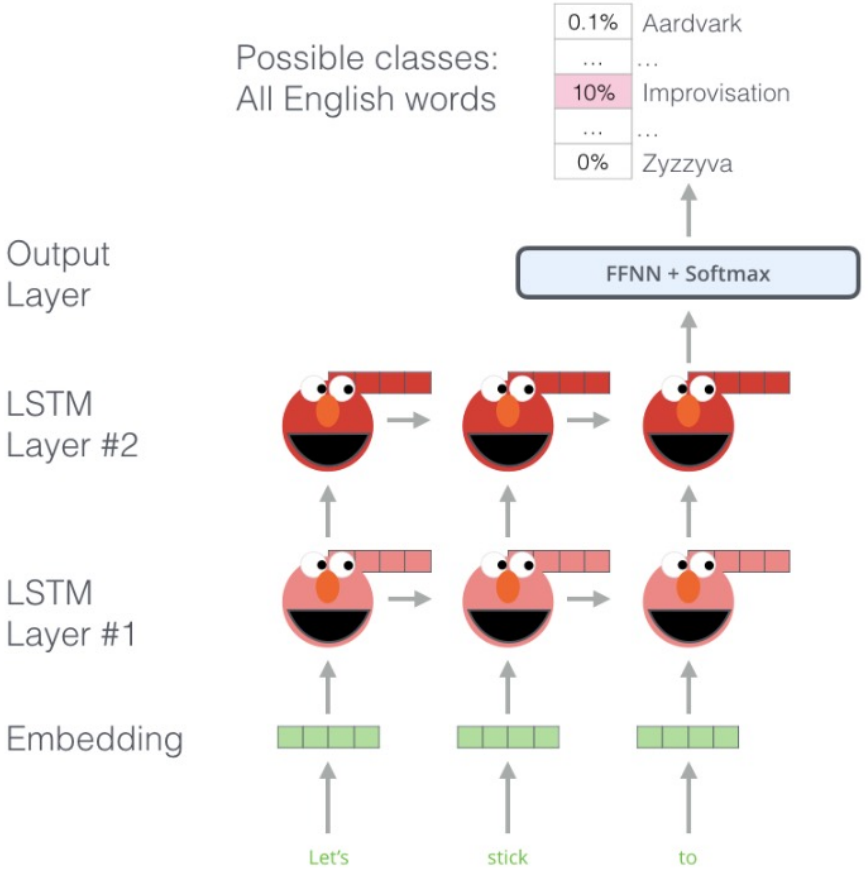
Bidirectional language models

Forward에서 해당 문장이 만들어질 확률

$$p(t_1, t_2, \ldots, t_N) = \prod_{k=1}^{N} p(t_k \mid t_1, t_2, \ldots, t_{k-1}).$$

Backward에서 해당 문장이 만들어질 확률

$$p(t_1, t_2, \ldots, t_N) = \prod_{k=1}^{N} p(t_k \mid t_{k+1}, t_{k+2}, \ldots, t_N).$$



Model 설명

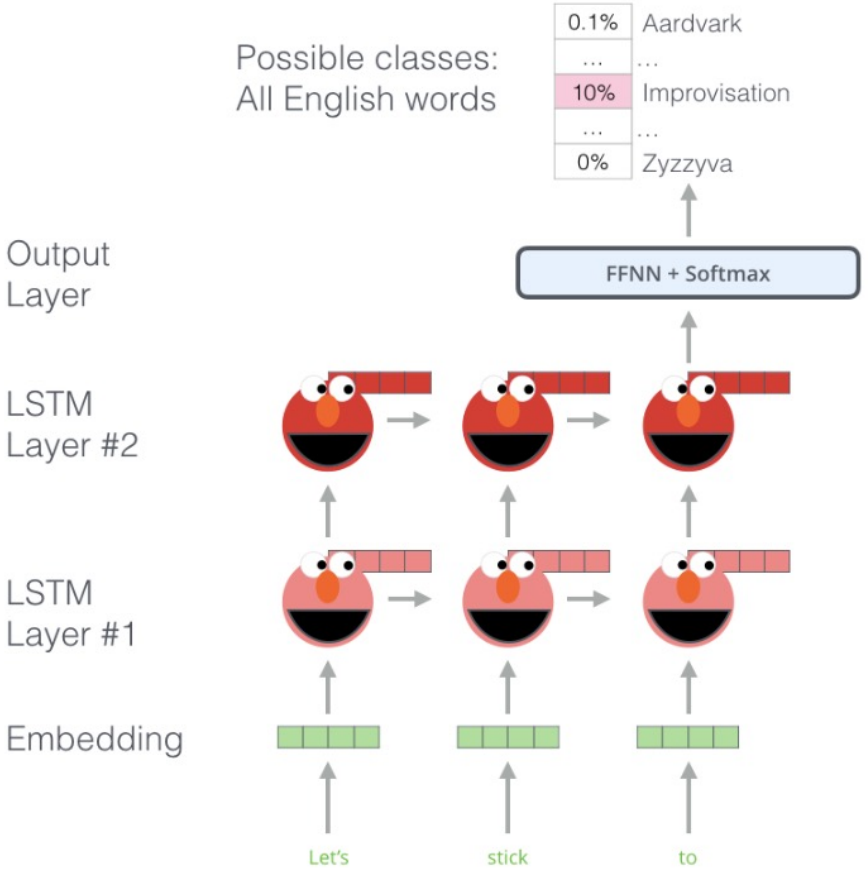
Bidirectional language models

이러한 Forward와 Backward의 각각의 parameter은 유지하면서 두 log likelihood를 maximize하는 것이 본 논문의 목적이다.

$$\sum_{k=1}^{N} \Big(\log p(t_k \mid t_1, \ldots, t_{k-1}; \Theta_x, \overrightarrow{\Theta}_{LSTM}, \Theta_s) + \log p(t_k \mid t_{k+1}, \ldots, t_N; \Theta_x, \overleftarrow{\Theta}_{LSTM}, \Theta_s) \Big) .$$

*Θ_x = token representation, Θ_{LSTM} = LSTM layer , Θ_s = Softmax layer 각 파라미터들

이때 Forward와 Backward의 각각의 parameter을 완전히 독립적으로 사용하는 것이 아닌, 가중치를 서로 다르게 줘서 parameter을 유지할 수 있었다.



TRAIN AND TEST

Model 설명

ELMo - pretrain

: biLM, 즉 언어 모델을 통해 앞뒤 맥락 기준으로 해당 자리의 단어를 예측 할 때에 언어모델 내부에서 추출된 특징을 원하는 Task에 맞게 가중치를 이용해 잘 조합해서 사용하는 아키텍쳐

- 각 토큰 마다 L-layer biLM(2L)과 처음의 token layer을 거쳐서 나오는 2L+1개의 representations

$$R_k = \{x_{LM_k}, \overrightarrow{h}_{LM_{k,j}}, \overleftarrow{h}_{LM_{k,j}} \mid j = 1, \ldots, L\} = \{h_{LM_{k,j}} \mid j = 0, \ldots, L\}$$

*j = layer index, k = token index

$$h_{LM_{k,j}} = [\overrightarrow{h}_{LM_{k,j}}; \overleftarrow{h}_{LM_{k,j}}]$$

- 다운스트림 모델에 포함하기 위해, R의 모든 층을 하나의 벡터로 축소

$$ELMo_k = E(R_k; \Theta_e)$$

$E(R_k) = h_{LM_{k,L}}$ 간단ver. 최상위 층의 vector

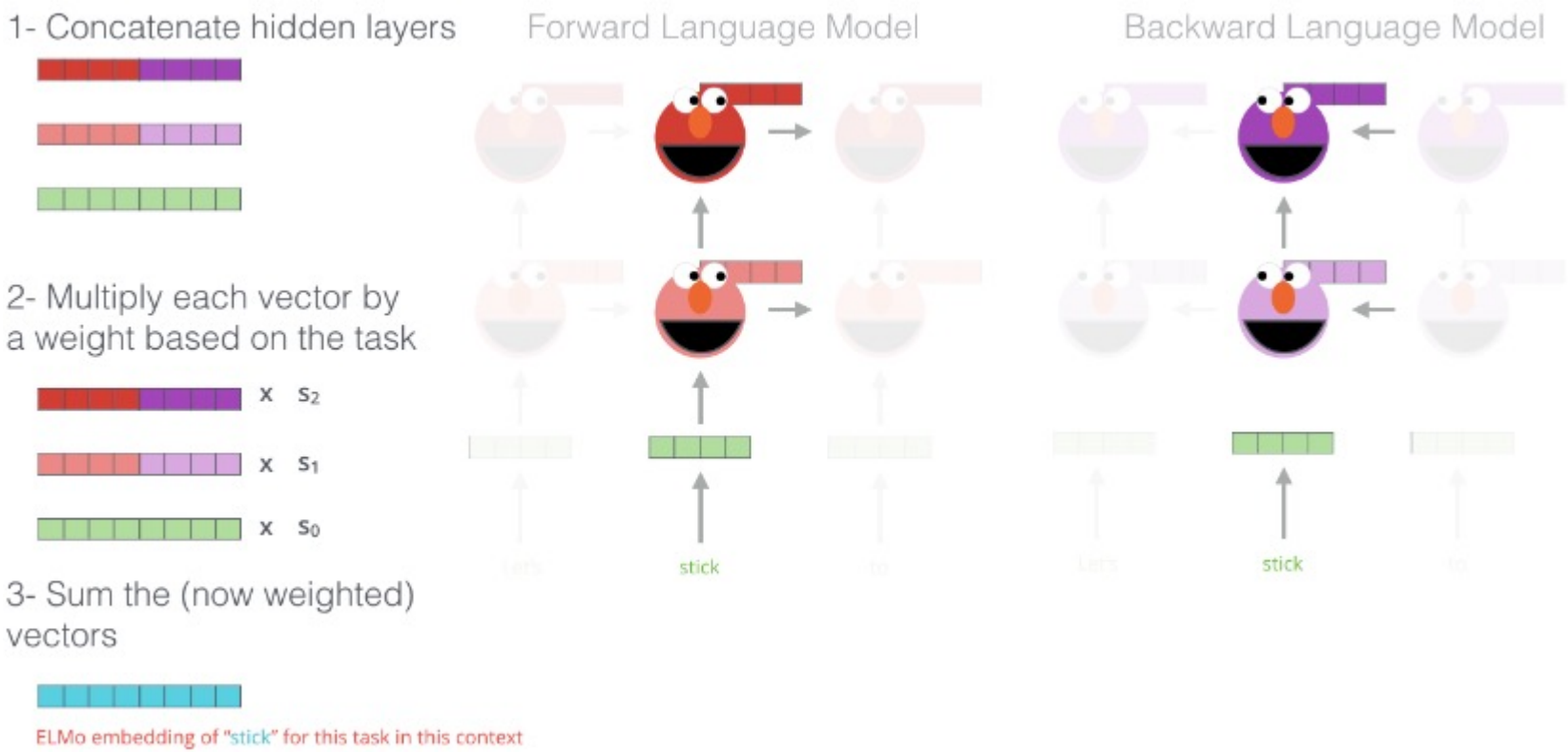
$$ELMo_{task_k} = E(R_k; \Theta_{task}) = \gamma_{task} \sum_{j=0}^{L} s_{task_j} h_{LM_{k,j}}$$

s는 Layer 층별 중요도를 학습하는 가중치로, softmax 가중치가 적용되며 모든 layer에 적용되는 값을 더하면 1이 됩니다.
감마의 경우 전체 모델의 적용되는 scale로 최적화에 아주 중요한 역할을 함

TRAIN AND TEST

Model 설명

ELMo - pretrain



TRAIN AND TEST

Model 설명

ELMo - task별 학습

: ELMo를 pretrain 하였으니 이제 task에 맞게 적용시켜 supervised architecture를 설계

1. biLM을 학습시켜 각 단어에 대한 layer representations를 저장하고(사전 학습),
2. 우리의 downstream이 pretrain 된 모델의 선형결합을 학습하도록 합니다.

Model 설명

ELMo - task별 학습

: ELMo를 pretrain 하였으니 이제 task에 맞게 적용시켜 supervised architecture를 설계

1. pretrained 된 biLM의 weights를 고정한다.

2. input text를 각 token에 대해 biLM을 거쳐 task에 맞게 $ELMO_k^{task}$ 를 뽑아낸다.

3. input token x_k^{LM}과 $ELMO_k^{task}$을 결합하여 $[x_k^{LM};ELMO_k^{task}]$ 형태의 input으로 Supervised task RNN모델에 넣어준다.
   - SNLI(문장 간 논리관계 유추), SQuAD(질의응답)의 task에는 출력에도$ELMO_k^{task}$를 두어 더 좋은 성능을 거둠

4. dropout과 loss에 biLM의 weights에 $\lambda||w||_2^2$를 더하는 정규화 방법도 사용.
   - inductive bias를 부과하여 모든 biLM layers에 평균에 근접하게 하기 위함(일반화 성능을 높이려고)

Model 설명

ELMo - task별 학습

: ELMo를 pretrain 하였으니 이제 task에 맞게 적용시켜 supervised architecture를 설계

Model 설명

Pre-trained bidirectional language model architecture

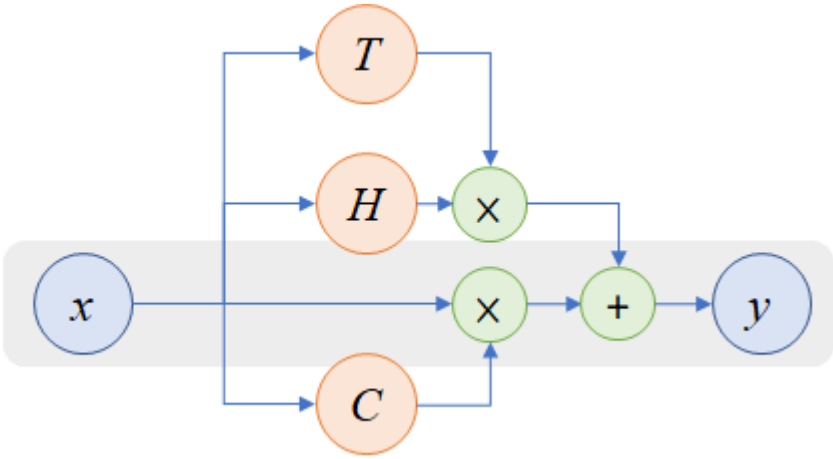
: 본 논문에서는 L = 2 biLSTM을 사용하였고, dimension = 512이며 LSTM 첫 번째와 두 번째 레이어 사이에 residual connection로 연결합니다. 임베딩은 2048 character n-gram convolutional filters에 두 개의 highway layer를 사용하였고 512차원으로 projection 시켜주었습니다.

residual connection:네트워크의 입력과 출력을 직접 연결하는 것, 역전파 동안 원래 입력의 그래디언트가 직접 전달되므로, 이 문제를 완화할 수 있다.

기존 임베딩 방법은 고정된 어휘 한 개의 representations을 생성하는 반면, biLM은 각 입력 token마다 순수 문자기반 입력 이기 때문에 3개의 representations을 생성합니다.

본 논문에서는 한 번의 pretrained으로 여러 task의 representations을 학습할 수 있고 특정 도메인별 데이터에서 biLM을 fine-tuning 하면 perplexity가 크게 감소하고, downstream의 성능이 향상된다는 장점을 설명한다

Information Highway



TRAIN AND TEST

Evaluation

Task	Previous SOTA		Our baseline	ELMo + baseline	Increase (absolute/ relative)
SQuAD	Liu et al. (2017)	84.4	81.1	85.8	4.7 / 24.9%
SNLI	Chen et al. (2017)	88.6	88.0	88.7 ± 0.17	0.7 / 5.8%
SRL	He et al. (2017)	81.7	81.4	84.6	3.2 / 17.2%
Coref	Lee et al. (2017)	67.2	67.2	70.4	3.2 / 9.8%
NER	Peters et al. (2017)	91.93 ± 0.19	90.15	92.22 ± 0.10	2.06 / 21%
SST-5	McCann et al. (2017)	53.7	51.4	54.7 ± 0.5	3.3 / 6.8%

TRAIN AND TEST

Evaluation

- Question Answering (질문 응답): 주어진 질문에 대해 주어진 문맥에서 올바른 답변을 찾아내는 작업입니다. 예를 들어, 위키피디아에서 추출된 문서에서 주어진 질문과 관련된 정보를 찾아 해당 질문에 대한 정확한 답변을 출력하는 것이 목표입니다.
- Textual Entailment (텍스트 추론): 주어진 전제 문장(또는 문장)과 가설 사이의 관계를 판단하여 가설이 전제를 기반으로 참인지 거짓인지를 결정하는 작업입니다. 즉, 전제가 주어졌을 때 가설이 그에 부합하는지를 판단하는 것이 목표입니다.
- Semantic Role Labeling (의미 역할 부착): 주어진 문장에서 동사와 그 인자들 간의 의미적 관계를 파악하는 작업입니다. 즉, 문장에서 각 단어의 역할을 정확히 레이블링하여 동사와 그 주체, 목적어 등의 관계를 파악하는 것이 목표입니다.
- Coreference Resolution (공지시 해소): 문장에서 여러 개체명이나 명사구가 특정 개체를 가리키는지를 파악하고, 이러한 여러 개체명이나 명사구를 해당하는 개체로 연결하는 작업입니다.
- Named Entity Recognition (개체명 인식): 주어진 텍스트에서 특정한 유형의 개체명(사람, 장소, 조직 등)을 식별하고 분류하는 작업입니다.
- Sentiment Analysis (감성 분석): 주어진 텍스트의 감정이나 태도를 파악하는 작업입니다. 예를 들어, 텍스트가 긍정적인지 부정적인지, 중립적인지를 분류하는 것이 목표입니다.

Analysis

- 본 논문에서는 모든 ELMo를 오직 input layer에만 적용하였지만 output layer에도 적용하였을 때 전반적으로 성능이 향상됨
- SQuAD, SNLI은 Input & Output 둘다 적용하였을때 성능이 가장 좋았고, SRL은 Input만 적용하였을때 성능이 좋았음. 여기서 SQuAD, SNLI은 biRNN이후 바로 attention layer를 사용하였기 때문에 이 layer에 ELMo를 추가하여 내부 representations에 직접 접근할 수 있도록 해주고, SRL의 경우 task-specific context representation이 biLM context representations 보다 더 중요하기 때문이라는 설명

Task	Input Only	Input & Output	Output Only
SQuAD	85.1	**85.6**	84.8
SNLI	88.9	**89.5**	88.7
SRL	**84.7**	84.3	80.9

TRAIN AND TEST

Analysis

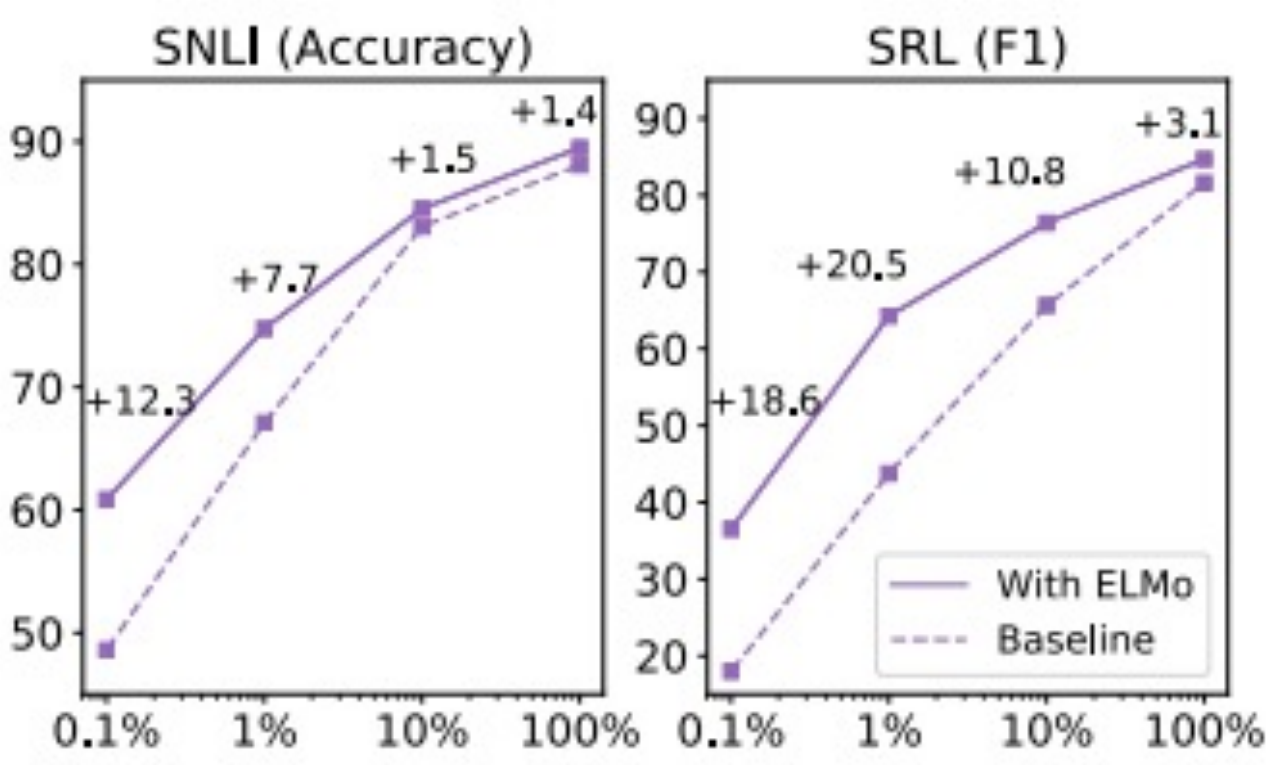
- ELMo를 적용한 task에 향상된 성능을 보이면서 biLM이 contextual representations를 잘 학습하여 NLP task에 유용하다는 것을 증명.
- biLM은 context로 word의 의미를 명확하게 해석

	Source	Nearest Neighbors
GloVe	play	playing, game, games, played, players, plays, player, Play, football, multiplayer
biLM	Chico Ruiz made a spectacular play on Alusik 's grounder {...}	Kieffer , the only junior in the group , was commended for his ability to hit in the clutch , as well as his all-round excellent play .
	Olivia De Havilland signed to do a Broadway play for Garson {...}	{...} they were actors who had been handed fat roles in a successful play , and had talent enough to fill the roles competently , with nice understatement .

TRAIN AND TEST

Analysis

- ELMo는 또한 샘플링의 효율성도 증가.
- 적은 train set으로도 ELMo를 적용한 모델이 좋은 성능을 도출



TRAIN AND TEST

TRAIN AND TEST