

ELMo(Embeddings from Language Models)란 무엇인가?

ELMo는 2018년 발표된 단어 임베딩 기법으로, 기존의 Word2Vec, GloVe 등과 달리 **문맥(Context)에 따라 단어 임베딩이 달라지는** 특징을 가집니다.

즉, 같은 단어라도 문장 내 위치나 주변 단어에 따라 임베딩 벡터가 다르게 생성됩니다.

기존의 단방향/양방향 LSTM과 ELMo의 차이

• 기존 LSTM

- 단방향 LSTM: 왼쪽(과거) → 오른쪽(미래) 방향으로만 정보를 전달
- 양방향 LSTM(BiLSTM): 왼쪽→오른쪽, 오른쪽→왼쪽 두 방향의 LSTM을 따로 학습하여, 두 결과를 단순히 합침(concatenate)
- 한계:
 - BiLSTM은 입력 전체를 한 번에 보고, 각 단어의 양방향 정보를 합치지만, 실제로는 각 방향의 LSTM이 독립적으로 동작함
 - 임베딩이 고정되어 있음(문맥에 따라 변하지 않음)

• ELMo

- **딥(Deep) 양방향 LSTM**을 사용
 - 여러 층의 BiLSTM을 쌓아 더 깊은 문맥 정보를 학습
- **문맥 기반 임베딩**
 - 각 단어의 임베딩이 문장 내 위치, 주변 단어에 따라 동적으로 생성됨
 - 예: "bank"가 "강가"일 때와 "은행"일 때 임베딩이 다름
- **특징**
 - 사전학습된 언어모델(대규모 텍스트 코퍼스에서 미리 학습)에서 각 층의 출력을 가중합하여 최종 임베딩으로 사용
 - 다운스트림 태스크(문장 분류, 개체명 인식 등)에 맞게 임베딩을 조정할 수 있음

왜 ELMo가 필요했는가? (왜 달라져야 하는가?)

• 고정 임베딩의 한계

- Word2Vec, GloVe 등은 단어별로 하나의 벡터만을 가짐(문맥 무시)
- 동음이의어, 다의어 구분 불가
 - 예: "나는 은행에 갔다"와 "강가의 은행"에서 "은행"의 의미가 다름에도 같은 벡터

• 문맥에 따라 의미가 달라지는 자연어의 특성

- 실제 언어에서는 단어의 의미가 주변 단어(문맥)에 따라 달라짐

- 문맥을 반영한 임베딩이 필요

- **성능 향상**

- 문맥 기반 임베딩을 사용하면, 개체명 인식(NER), 문장 분류 등 다양한 자연어처리 태스크에서 성능이 크게 향상됨

요약

- ELMo는 **딥 바이디렉셔널 LSTM**을 활용하여, 각 단어의 임베딩을 문맥에 따라 동적으로 생성하는 기법입니다.
- 기존의 (Bi)LSTM 기반 임베딩과 달리, **문맥 정보를 깊이 있게 반영**하여 자연어처리 성능을 크게 높였습니다.
- 이후 등장한 BERT, GPT 등 트랜스포머 기반 모델의 문맥 임베딩 개념의 시초가 되었습니다.