



# ELMo

## Introduction

- Word2Vec, GloVe 같은 방법은 **단어별로 고정된 벡터**를 생성  
→ 하지만 단어는 문맥에 따라 의미가 달라질 수 있으므로 이는 **문법 구조에 따른 의미 변형 혹은 다의어 문제**를 극복하지 못함
- Word2Vec과 같은 기존 static한 임베딩 방식의 한계를 극복하고, 문맥에 따라 단어 의미를 다르게 표현하는 **문맥적 단어 표현** 개념을 도입 → **Contextualized Word Representations**

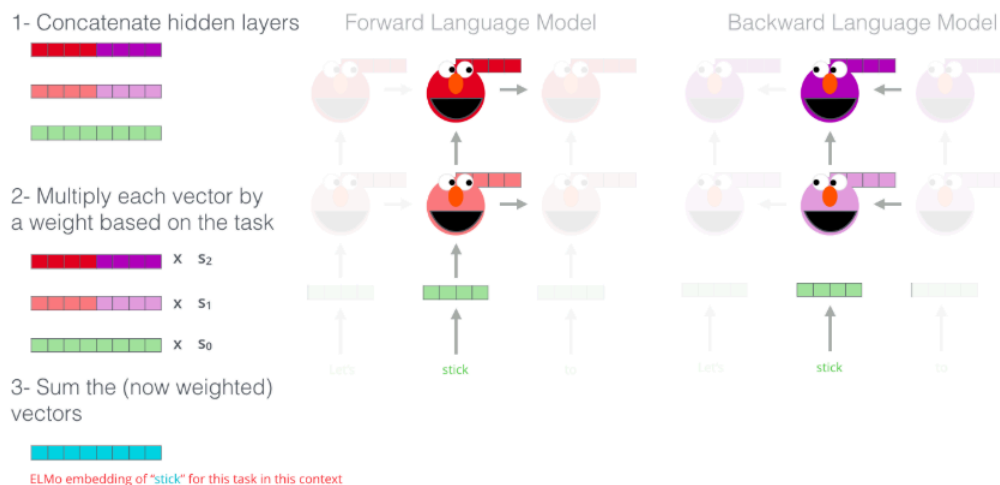
## Model Architecture

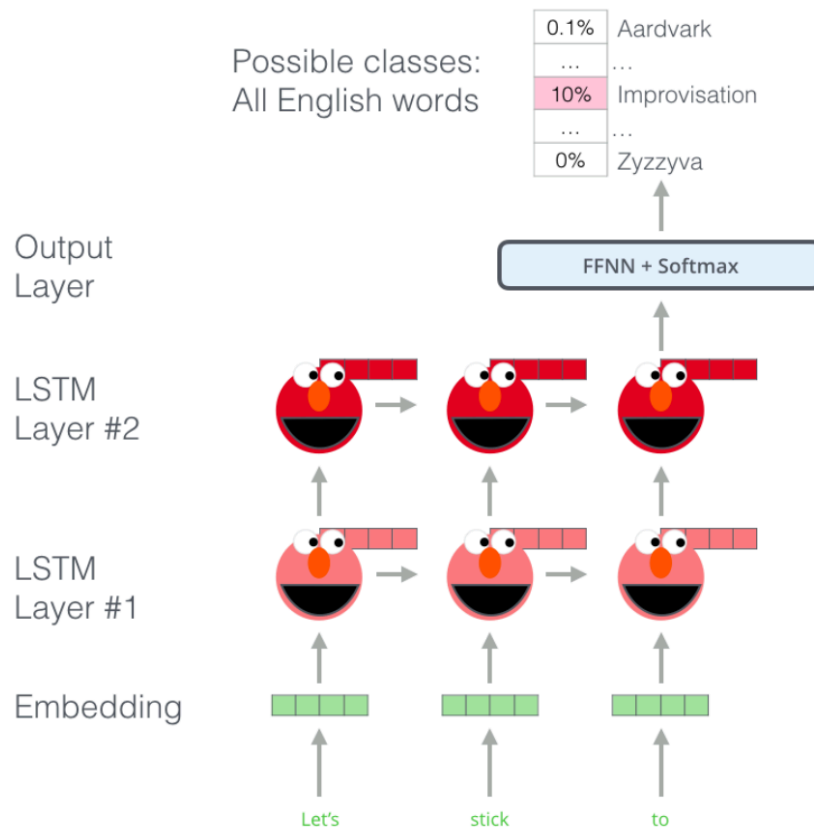
: BiLSTM 기반의 사전 학습된 신경망 언어 모델에서 추출된 벡터를 활용

→ BiLSTM이 앞뒤 정보를 모두 반영하여 단어 의미를 조정할 수 있음

- forward LM은 현재 토큰을 기준으로 다음에 올 토큰을 예측하는 역할, 그리고 backward LM은 현재 토큰 기준으로 이전의 토큰들을 예측하는 역할 → 이로 문맥이 반영될 수 있음

Embedding of "stick" in "Let's stick to" - Step #2





: Character-level CNN → BiLSTM (2-layer: Forward+Backward) → Weighted Sum of Representations (따라서 이때 각 단어 벡터는 다양한 층의 출력을 가중 합산하여 표현됨)

- 초기 임베딩을 Character Based CNN으로 각 단어→ 개별 문자 단위로 encoding → 이를 고정된 차원의 벡터로 표현함 (Out-of-Vocabulary 문제가 해결)

$$ELMo_k = \gamma \sum_{i=0}^L s_i h_k^i$$

LSTM의 여러 층에서 나온 벡터를 가중합하여 문맥에 따라 단어의 벡터가 달라질 수 있음

## Results

- 다양한 자연어 처리 벤치마크에서 큰 성능 향상을 보임 + 이때 input+output layer에 모두 도입했을 때 비교적 성능이 우수

## Future

- BiLSTM 기반이라 Transformer 기반 모델 GPT, BERT에 비해 연산량이 많음

- 또한 이때 단어 수준의 문맥 정보는 반영하지만, 전체 문장의 관계를 학습하지 않을 수 있음