Title: ELMo

URL: https://en.wikipedia.org/wiki/ELMo

PageID: 64781650

Categories: Category:Computational linguistics, Category:Machine learning, Category:Natural language processing, Category:Natural language processing software

---

ELMo (embeddings from language model) is a word embedding method for representing a sequence of words as a corresponding sequence of vectors. It was created by researchers at the Allen Institute for Artificial Intelligence , and University of Washington and first released in February 2018. It is a bidirectional LSTM which takes character-level as inputs and produces word-level embeddings, trained on a corpus of about 30 million sentences and 1 billion words.

The architecture of ELMo accomplishes a contextual understanding of tokens . Deep contextualized word representation is useful for many natural language processing tasks, such as coreference resolution and polysemy resolution.

ELMo was historically important as a pioneer of self-supervised generative pretraining followed by fine-tuning , where a large model is trained to reproduce a large corpus, then the large model is augmented with additional task-specific weights and fine-tuned on supervised task data. It was an instrumental step in the evolution towards transformer-based language modelling.

Architecture

ELMo is a multilayered bidirectional LSTM on top of a token embedding layer. The output of all LSTMs concatenated together consists of the token embedding.

The input text sequence is first mapped by an embedding layer into a sequence of vectors. Then two parts are run in parallel over it. The forward part is a 2-layered LSTM with 4096 units and 512 dimension projections, and a residual connection from the first to second layer. The backward part has the same architecture, but processes the sequence back-to-front. The outputs from all 5 components (embedding layer, two forward LSTM layers, and two backward LSTM layers) are concatenated and multiplied by a linear matrix ("projection matrix") to produce a 512-dimensional representation per input token.

ELMo was pretrained on a text corpus of 1 billion words. The forward part is trained by repeatedly predicting the next token, and the backward part is trained by repeatedly predicting the previous token.

After the ELMo model is pretrained, its parameters are frozen, except for the projection matrix, which can be fine-tuned to minimize loss on specific language tasks. This is an early example of the pretraining-fine-tune paradigm . The original paper demonstrated this by improving state of the art on six benchmark NLP tasks.

Contextual word representation

The architecture of ELMo accomplishes a contextual understanding of tokens. For example, the first forward LSTM of ELMo would process each input token in the context of all previous tokens, and the first backward LSTM would process each token in the context of all subsequent tokens. The second forward LSTM would then incorporate those to further contextualize each token.

Deep contextualized word representation is useful for many natural language processing tasks, such as coreference resolution and polysemy resolution. For example, consider the sentence

She went to the bank to withdraw money.

In order to represent the token "bank", the model must resolve its polysemy in context.

The first forward LSTM would process "bank" in the context of "She went to the", which would allow it to represent the word to be a location that the subject is going towards.

The first backward LSTM would process "bank" in the context of "to withdraw money", which would allow it to disambiguate the word as referring to a financial institution.

The second forward LSTM can then process "bank" using the representation vector provided by the first backward LSTM, thus allowing it to represent it to be a financial institution that the subject is going towards.

Historical context

ELMo is one link in a historical evolution of language modelling. Consider a simple problem of document classification , where we want to assign a label (e.g., "spam", "not spam", "politics", "sports") to a given piece of text.

The simplest approach is the " bag of words " approach, where each word in the document is treated independently, and its frequency is used as a feature for classification. This was computationally cheap but ignored the order of words and their context within the sentence. GloVe and Word2Vec built upon this by learning fixed vector representations (embeddings) for words based on their co-occurrence patterns in large text corpora.

Like BERT (but unlike "bag of words" such as Word2Vec and GloVe), ELMo word embeddings are context-sensitive, producing different representations for words that share the same spelling. It was trained on a corpus of about 30 million sentences and 1 billion words. Previously, bidirectional LSTM was used for contextualized word representation. ELMo applied the idea to a large scale, achieving state of the art performance.

After the 2017 publication of Transformer architecture , the architecture of ELMo was changed from a multilayered bidirectional LSTM to a Transformer encoder, giving rise to BERT . BERT has a similar pretrain-fine-tune workflow, but uses a Transformer with implications for more parallelizable training.

References