动态词向量算法 — ELMo

原创 NLP与人工智能 NLP与人工智能 2019-11-09

> ELMo
>
> 传统的词向量模型，例如 Word2Vec 和 Glove 学习得到的词向量是固定不变的，即一个单词只有一种词向量，显然不适合用于多义词。而 ELMo 算法使用了深度双向语言模型(biLM)，只训练语言模型，而单词的词向量是在输入句子实时获得的，因此词向量与上下文信息密切相关，可以较好地区分歧义。

静态词向量算法

在之前的文章中介绍了词嵌入算法 Word2Vec 和 Glove。与传统的 one-hot 编码、共现向量相比，词嵌入算法得到的词向量维度更低、也可以比较好地支持一些下游的任务，例如文档分类，问答系统等。

但是这两种算法都是**静态词向量**算法，在数据集上训练好一个语言模型之后，每一个词的词向量就固定下来了。后续使用词向量时，无论输入的句子是什么，词向量都是一样的，例如：

- “我喜欢吃小米”中的“小米”指一种食物
- “小米手机挺好用”中的“小米”指手机品牌

给定上面两个句子，在 Word2Vec 和 Glove 中去得到“小米”的词向量都是一样的，不能根据上下文给出更准确的词向量。

而 **ELMo** 是一种动态词向量算法，在大型的语料库里训练一个 biLSTM (双向LSTM模型)。下游任务需要获取单词词向量的时候，将整个句子输入 biLSTM，利用 biLSTM 的输出作为单词的词向量，包含了上下文信息。可以理解成，biLSTM 是一个函数，函数的输入是一个句子，输出是句子中单词的词向量

双向语言模型

首先介绍什么是双向语言模型，以及如何通过 biLSTM 得到单词的词向量，对 LSTM 不熟悉的童鞋可以参考前一篇文章《循环神经网络 RNN、LSTM、GRU》。

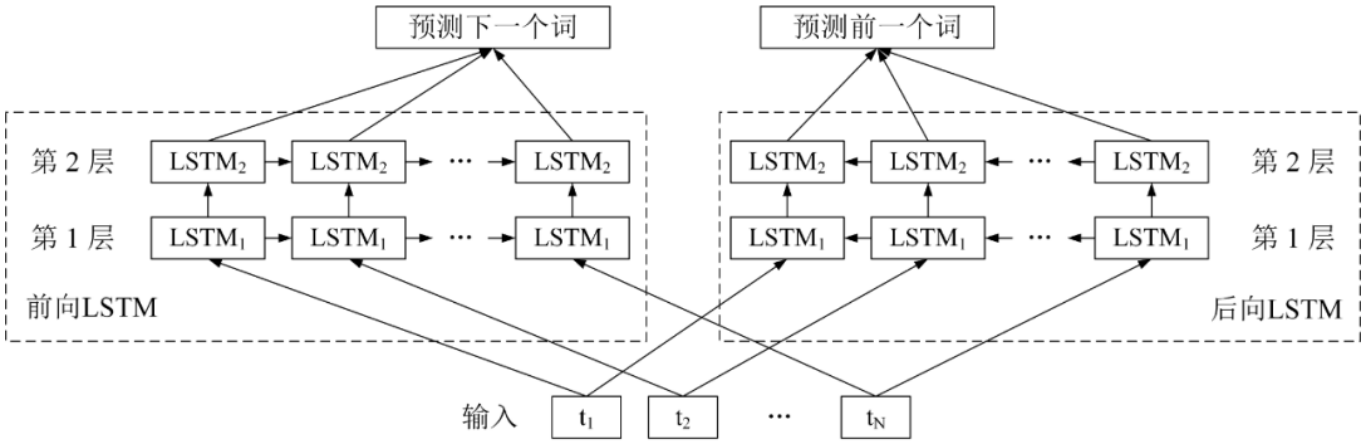
双向语言模型

双向语言模型包括**前向模型**和**后向模型**，给定一个包含 N 个单词的句子 T = [t(1), t(2), ..., t(N)]，**前向模型**需要通过前面的单词 [t(1), t(2), ..., t(k-1)] 预测下一个单词 t(k)，而后向模型需要通过后面的单词，预测前一个单词。

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

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

使用 biLSTM 得到上下文相关的词向量



biLSTM 是一种双向的循环神经网络，包含了前向网络与后向网络两部分。上图是一个层数 L=2 的 biLSTM。

每一个单词 t(i) 的输入是词向量，这个词向量是固定的，可以使用 Word2Vec 或者 Glove 生成的词向量，在 ELMo 中使用了 **CNN-BIG-LSTM** 生成的词向量。**注意，ELMo 输入时的词向量是固定的，ELMo 将输入的词向量传到 biLSTM 得到的才是动态的，包含上下文信息。**

ELMo 的论文中使用以下符号表示**双向 LSTM 中每一层对应第 i 个单词的输出**。其中**前向**输出包含第 i 个单词之前的语义，**后向**输出包含了第 i 个单词之后的语义。

$\vec{h}_{k,j}^{LM}$ 第k个输入单词在第j层 前向 LSTM 的输出

$\overleftarrow{h}_{k,j}^{LM}$ 第k个输入单词在第j层 后向 LSTM 的输出

$\vec{h}_{k,j}^{LM}$ 包含前文信息，$\overleftarrow{h}_{k,j}^{LM}$包含后文信息

文章中比较难添加公式，因此使用 **h**(k,j,→) 表示前向输出，使用 **h**(k,j,←) 表示后向向输出，请谅解。

每一层的输出 h(k-1,j,→) 和 h(k+1,j,←) 都是单词的动态词向量。

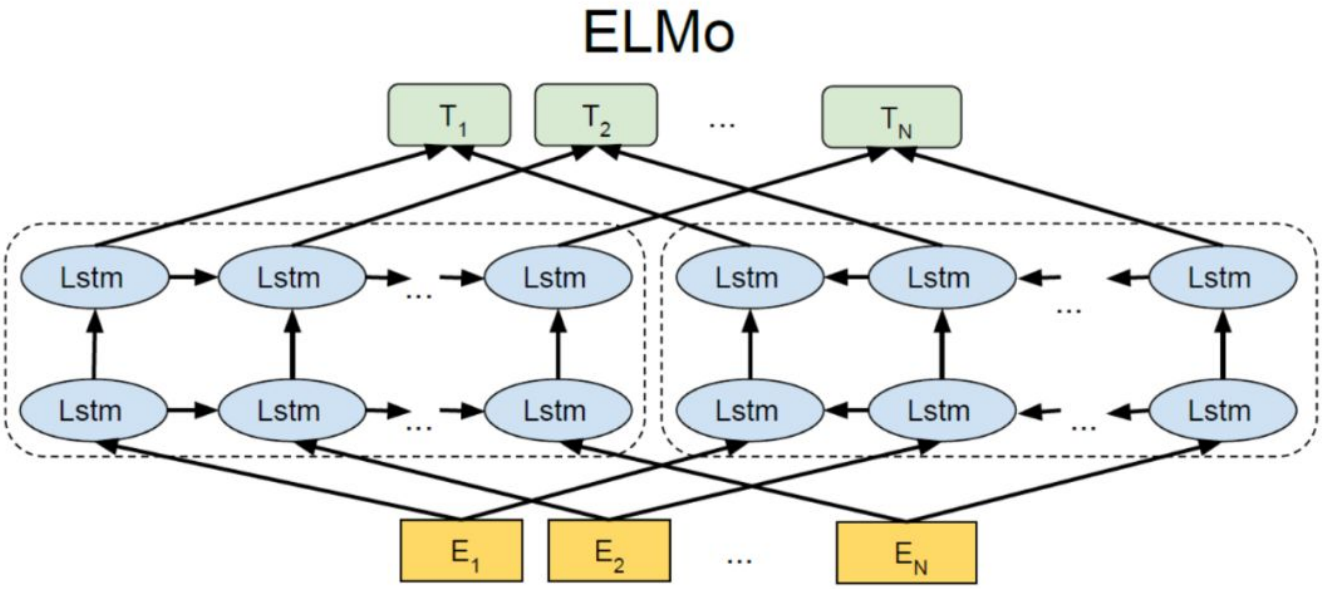
LSTM 一共 L 层，对于前向 LSTM，每一个单词 t(k-1) 的最后一层输出 **h**(k-1,L,→) 用于预测下一个单词 t(k)；对于后向 LSTM，每一个单词 t(k+1) 的最后一层输出 **h**(k+1,L,←) 用于预测前一个单词 t(k)。预测的过程采用 softmax， biLSTM 需要优化的目标函数如下:

$$\sum_{k=1}^{N}\left(\log p\left(t_k|t_1,\ldots,t_{k-1};\theta_x,\vec{\theta}_{LSTM},\theta_s\right)\right)$$
$$+\left(\log p\left(t_k|t_{k+1},\ldots,t_N;\theta_x,\overleftarrow{\theta}_{LSTM},\theta_s\right)\right)$$

θ(x) 表示单词输入时候的词向量，这个词向量是固定的。**θ**(s) 表示 softmax 层，用于预测前后的单词。**θ**(LSTM,→) 表示前向 LSTM 的参数，用于计算 **h**(k-1,L,→)。**θ**(LSTM,←) 表示后向 LSTM 的参数，用于计算 **h**(k-1,L,←)。

ELMo 算法

流程介绍



ELMo 中使用的 biLSTM 层数 L=2，ELMo 首先在大型的数据集上训练好模型，然后再后续任务中可以根据输入的句子，输出每一个单词的词向量。例如给定一个句子 T=[t(1), t(2), ..., t(N)]，ELMo 计算词向量的方法如下：

- 从静态的词向量表里查找单词的词向量 E(1), ..., E(N) 用于输入。ELMo 使用 CNN-BIG-LSTM 生成的词向量作为输入。

- 将单词词向量 E(1), ..., E(N) 分别输入第 1 层前向 LSTM 和后向 LSTM，得到前向输出 **h**(1,1,→), ..., **h**(N,1,→) 和后向输出 **h**(1,1,←), ..., **h**(N,1,←)。

- 将前向输出 **h**(1,1,→), ..., **h**(N,1,→) 传入到第 2 层前向 LSTM，得到第 2 层前向输出 **h**(1,2,→), ..., **h**(N,2,→)；然后将后向输出 **h**(1,1,←), ..., **h**(N,1,←) 传入到第 2 层后向 LSTM，得到第 2 层后向输出 **h**(1,2,←), ..., **h**(N,2,←)。

- 则单词 i 最终可以得到的词向量包括 E(i), **h**(N,1,→), **h**(N,1,←), **h**(N,2,→), **h**(N,2,←)，如果采用 L 层的 biLSTM 则最终可以得到 2L+1 个词向量。

使用词向量

在上面我们知道句子中一个单词 i 可以得到 2L+1 个词向量，在实际使用的过程中应该如何利用这 2L+1 个词向量?

首先在 ELMo 中使用 CNN-BIG-LSTM 词向量 E(i) 作为输入，E(i) 的维度等于 512。然后每一层 LSTM 可以得到两个词向量 **h**(i,layer,→) 和 **h**(i,layer,←)，这两个向量也都是 512 维。则对于单词 i 可以构造出 L+1 个词向量。

$$h_{i,j}^{LM} \quad j = 0,1,\ldots,L$$

$$h_{i,0}^{LM} = [E_i; E_i] \text{ 表示输入的词向量}$$

$$h_{i,j}^{LM} = [\overrightarrow{h}_{i,j}^{LM}; \overleftarrow{h}_{i,j}^{LM}] \text{ 表示第 j 层 biLSTM 的输出词向量}$$

h(i,0) 表示两个 E(i) 直接拼接，表示输入词向量，这是静态的，1024 维。

h(i,j) 表示第 j 层 biLSTM 的两个输出词向量 **h**(i,j,→) 和 **h**(i,j,←) 直接拼接，这是动态的，1024维。

ELMo 中不同层的词向量往往的侧重点往往是不同的，输入层采用的 CNN-BIG-LSTM 词向量可以比较好编码词性信息，第 1 层 LSTM 可以比较好编码句法信息，第 2 层 LSTM 可以比较好编码单词语义信息。

ELMo 的作者提出了两种使用词向量的方法：

第一种是直接使用最后一层 biLSTM 的输出作为词向量，即 **h**(i,L)。

第二种是更加通用的做法，将 L+1 个输出加权融合在一起，公式如下。γ 是一个与任务相关的系数，允许不同的 NLP 任务缩放 ELMo 的向量，可以增加模型的灵活性。s(task,j) 是使用 softmax 归一化的权重系数。

$$ELMo_i^{task} = \gamma^{task} \sum_{j=0}^{L} s_j^{task} h_{i,j}^{LM}$$

ELMo效果

	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 .

这是论文中的一个例子，上面的是 Glove，下面两行是 ELMo。

可以看到 Glove 查找 play 的最近邻，会出现“游戏”、“表演”、“运动”等相关的单词，可能与 paly 在句子中的实际意思不同。

但是在 ELMo 中，可以看到第一个句子中的 play 是比赛的意思，其最近邻句子的 play 也是比赛的意思。而第二个句子的 play 都是表演的意思。说明 ELMo 可以根据上下文更好地得到一个单词的词向量。

ELMo 总结

- ELMo 训练语言模型，而不是直接训练得到单词的词向量，在后续使用中可以把句子传入语言模型，结合上下文语义得到单词更准确的词向量。
- 使用了 biLSTM，可以同时学习得到保存上文信息和下文信息的词向量。
- biLSTM 中不同层得到的词向量侧重点不同，输入层采用的 CNN-BIG-LSTM 词向量可以比较好编码词性信息，第 1 层 LSTM 可以比较好编码句法信息，第 2 层 LSTM 可以比较好编码单词语义信息。通过多层词向量的融合得到最终词向量，最终词向量可以兼顾多种不同层次的信息。

参考文献

1. Deep contextualized word representations
https://arxiv.org/pdf/1810.04805.pdf
2. 知乎：ELMo原理解析及简单上手使用
https://zhuanlan.zhihu.com/p/51679783