Matching the Blanks: Distributional Similarity for Relation Learning

Livio Baldini Soares **Nicholas FitzGerald** **Jeffrey Ling*** **Tom Kwiatkowski**
Google Research
{liviobs,nfitz,jeffreyling,tomkwiat}@google.com

Google的ACL 2019
很有google特色。。。

其实我的理解，就是一个RE上的基于BERT的pre-train model。
paper主要包括：

- 探究怎么用bert来adapt到这个task合适
- 如何unsupervised的train
- experiment

然后探究的task有：

- supervised
- few-shot

两种task对应的模式结构：

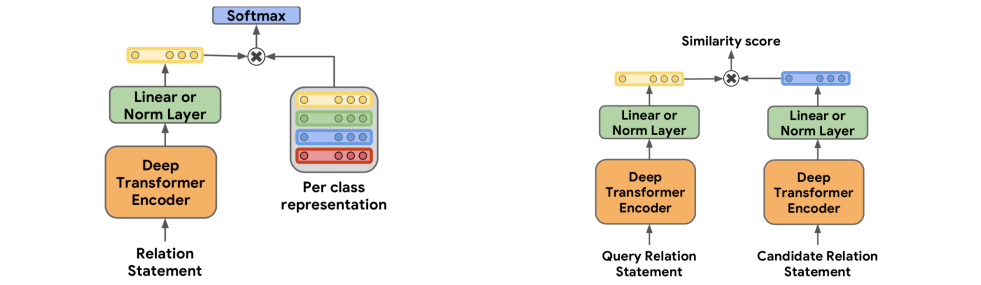


Figure 2: Illustration of losses used in our models. The left figure depicts a model suitable for supervised training, where the model is expected to classify over a predefined dictionary of relation types. The figure on the right depicts a pairwise similarity loss used for few-shot classification task.

这个也很好理解，左边那个有监督信号，所以是一个relation classification的问题。
右边是一个N way K shot的问题，通过比较与support set中哪个最相似就划入哪一个class。
还有一个需要说明的是，这个task是给定一个sentence，里面标好了head和

tail。
也就是说，**一个sentence只有一个head和一个tail。而不是之前那种一个sentence有多个relation的**。

（针对有多个relation的情况，不是还出现了：

- 一个entity/word属于多个relation的情况的解决，如HBT、CopyRE之类的
- 如何利用已经relation来constuct 新的mutli-hot relation如 A walk-based model *** ）

所以对于这个一个sentence一个(h,t)的task，paper就想用一种把这个关于entity-pair的statement/sentence 编码然后分类的方式来进行。
听起来是不是简单粗暴，因为他的encoder用的是BERT。

paper的第一部分，探究哪种方式用BERT最好。
探究了下面六种方式：

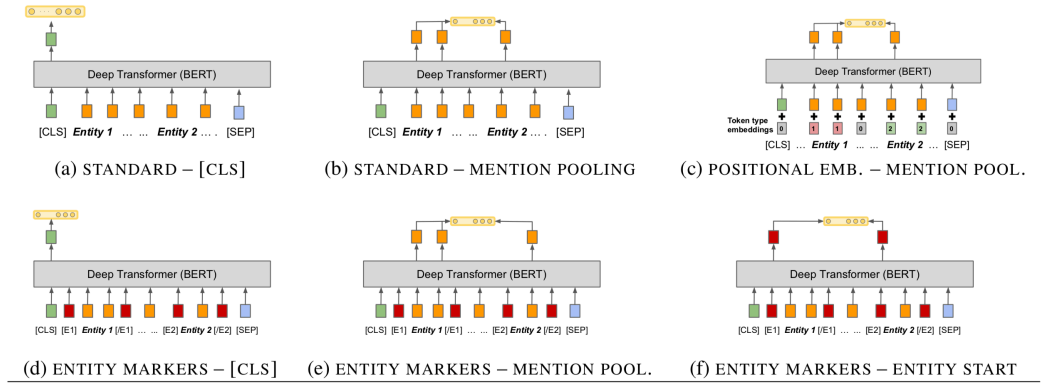


Figure 3: Variants of architectures for extracting relation representations from deep Transformers network. Figure (a) depicts a model with STANDARD input and [CLS] output, Figure (b) depicts a model with STANDARD input and MENTION POOLING output and Figure (c) depicts a model with POSITIONAL EMBEDDINGS input and MENTION POOLING output. Figures (d), (e), and (f) use ENTITY MARKERS input while using [CLS], MENTION POOLING, and ENTITY START output, respectively.

主要包括输入方式：

- standard的输入方式
- 加上positional embedding，这个不是standard的那种，是给entity1加入一个，entity2加入一个
- entity marker：在entity1/2的前后分别加一个标志符，explicitly地标出来entity位置

还有输出方式：

- 用[cls]的对应输出，这也是BERT很多fine-tune的采用的方式
- mention pooling：entity的词对应的output去pooling
- entity start：用entity开头的那个标志符

最后分别在task上用上述模型结构进行fine-tune，得到结论：**使用entity marker + entity_start的效果最好**。

接下来**paper的第二部分，是提出了一个无监督预训练的方**式。
就像BERT在MLMtask上pre-train一样。
作者首先提出一个assumption：**包含同样实体对的句子，他们编码起来之后向量表达的意思应该差不多，因此他们的内积应该尽量大。反之应该尽量小**。
具体地分类通过

$$p(l=1|\mathbf{r},\mathbf{r}')=\frac{1}{1+\exp f_\theta(\mathbf{r})^\top f_\theta(\mathbf{r}')}$$

然后呢就通过下面这个loss来：

$$\mathcal{L}(\mathcal{D})=-\frac{1}{|\mathcal{D}|^2}\sum_{(\mathbf{r},e_1,e_2)\in\mathcal{D}}\sum_{(\mathbf{r}',e_1',e_2')\in\mathcal{D}} \tag{1}$$
$$\delta_{e_1,e_1'}\delta_{e_2,e_2'}\cdot\log p(l=1|\mathbf{r},\mathbf{r}')+$$
$$(1-\delta_{e_1,e_1'}\delta_{e_2,e_2'})\cdot\log(1-p(l=1|\mathbf{r},\mathbf{r}'))$$

where $\delta_{e,e'}$ is the Kronecker delta that takes the value 1 iff $e=e'$, and 0 otherwise.

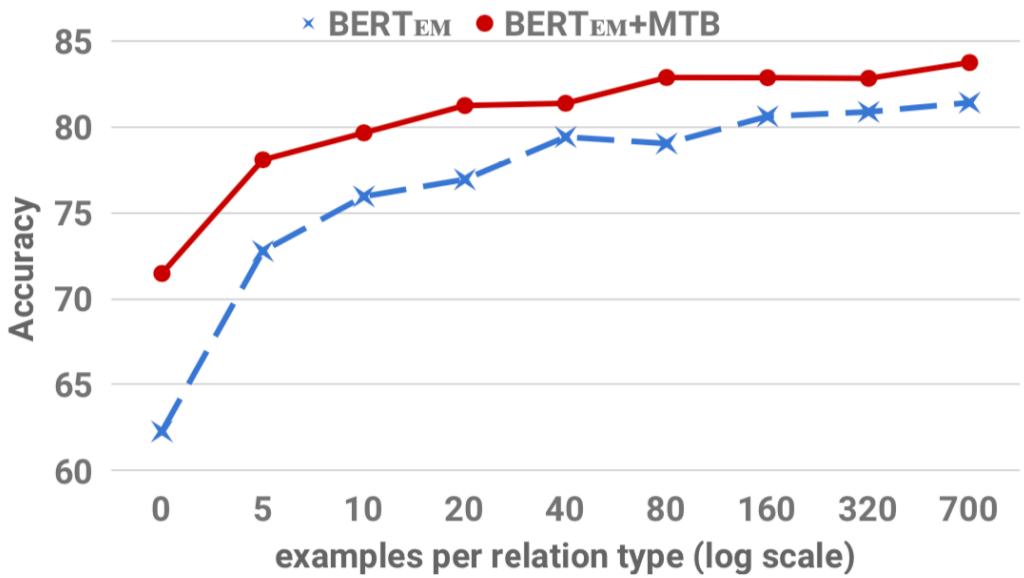
当然了还没结束，还需要继续优化改进，改进主要包括两方面。
第一个，如果直接原句子搞进去，那么模型很可能就是通过看两个entity来学

习，完全忽视了其他context。
而这个context其实才是relation的重点。尽管同样entity-pair的句子是有很大可能表达完全一样的意思，但是这个意思的本质还是由statement来represent。
所以，类似BERT的[mask]，这里paper采用了将entity随机mask掉的策略。具体的，
每个entity都有alpha（实验取0.7）概率保留原entity，否则替换成[BLANK].
第二个就是防止训练数据的bias问题：

- 对于正例，含有同样entity-pair的句子，采样同样数量的句子
- 对于负例，不可能去遍历所有负例的，所以采用negative sampling。主要包括两个entity都不同，以及一个相同一个不同的负例。

之后就是在大量语料上进行pre-train啦。
之后就是做实验啦。

首先对于FewRel的few-shot实验，模型在没有见到训练数据的情况下，就已经outperform其他方法达到sota了。。。
然后又加了一些训练数据去fine-tune。



对于supervised RE的也是类似的效果。