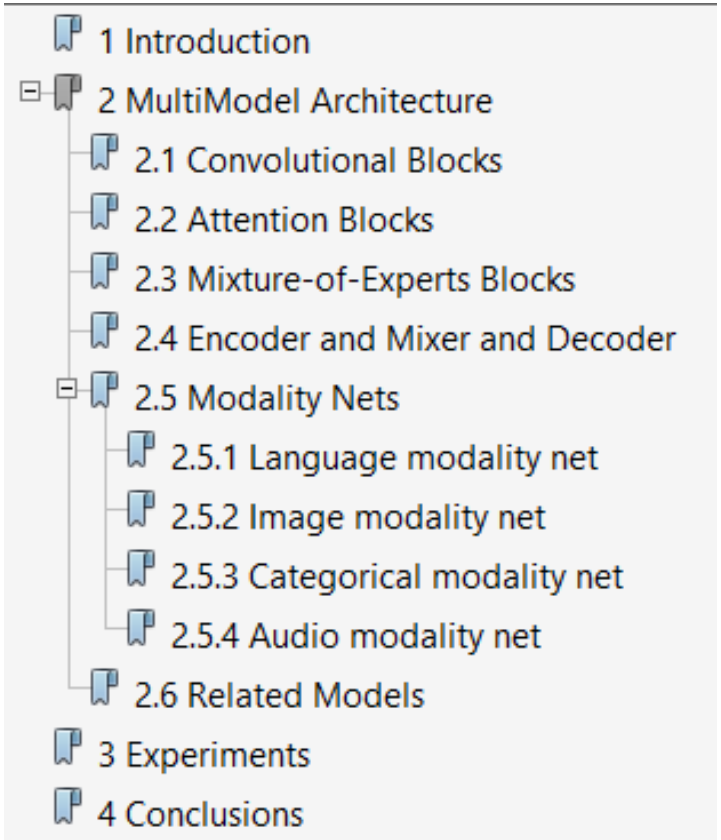
One Model to learn them All

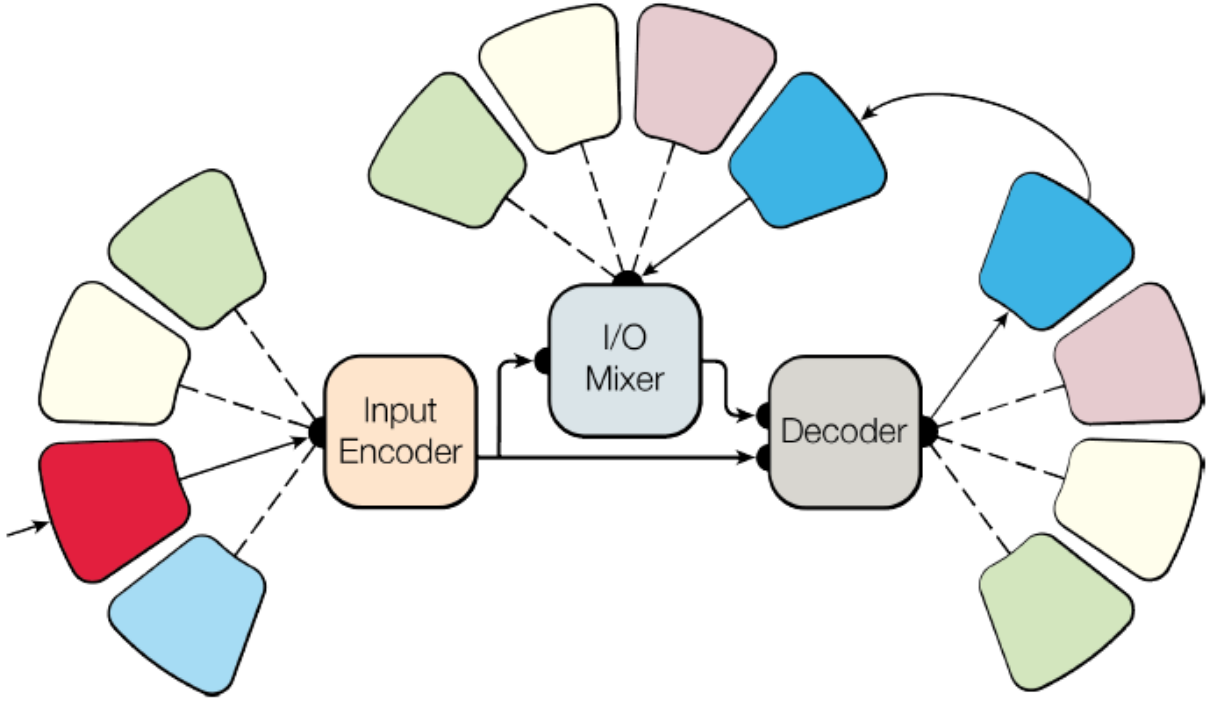
内容摘要：

在此之前，深度学习中的模型完成的任务都是单一的（图片，语音，文本....），对某一类任务要单独设计训练模型，这无疑增加了工作量。基于此作者提出了一种模型MultiModel，这种模型可以跨领域产生比较好的效果，数据也是同时训练的。这种模型集合了多个领域：卷积层，注意力机制，稀疏门控专家混合层。



Structure

- 模式网：文本，图像，音频
- 编码器
- I/O混合器
- 解码器



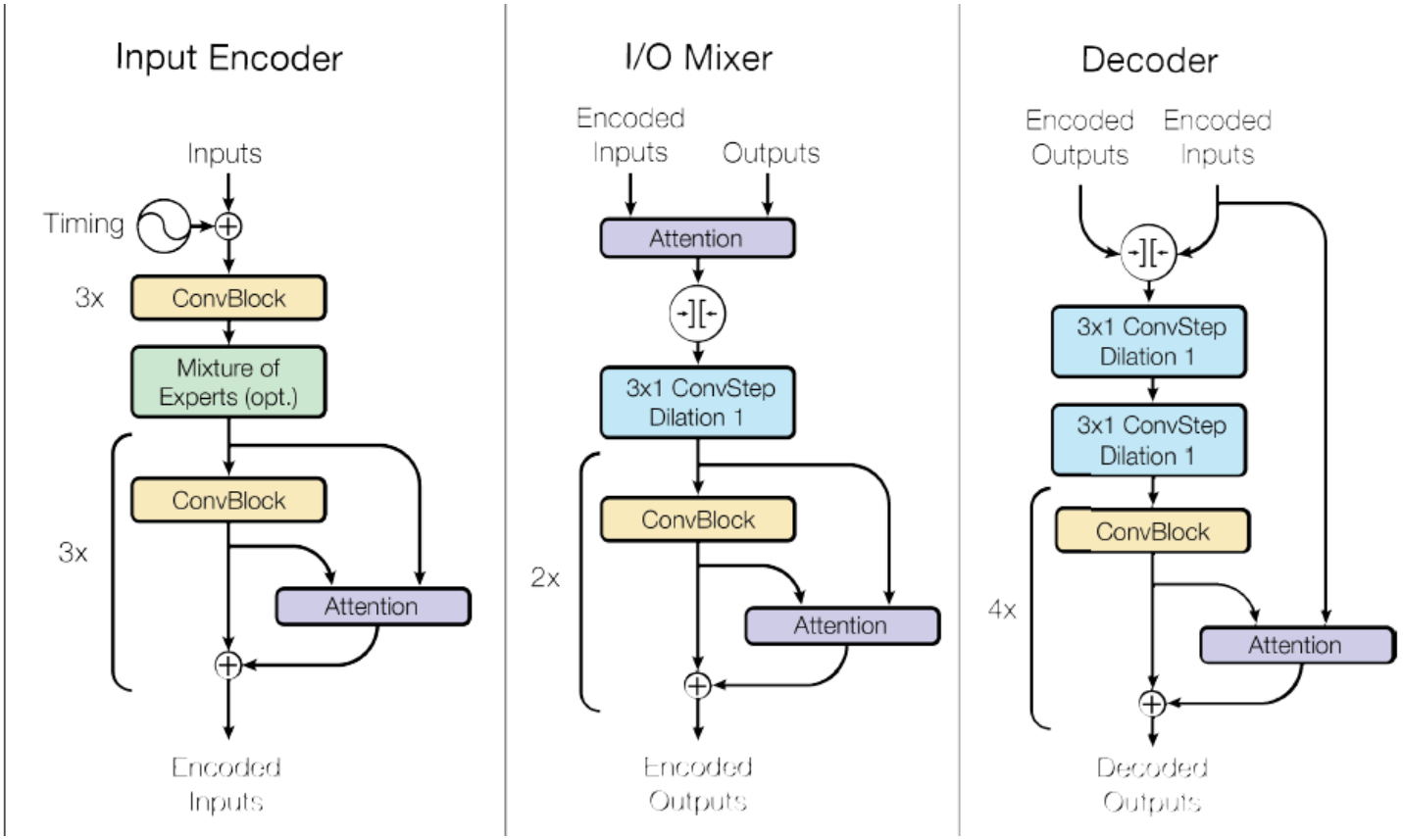
Encoder和Decoder通过三个关键的计算模块完成：

1. 卷积层：提取局部特征，在整个空间泛化；
2. 注意层：关注特定因素，提高模型性能；
3. 稀疏门控混合专家层：让模型避免过多的计算成本。

Structure—Encoder and Mixer and Decoder

这三种模型与全卷积的序列到序列模型ByteNet、WaveNet类似，不同的是其中的计算模块。结构如右图所示。

为了完成多种任务，Decoder的输入伴随着一个嵌入式的指令标记向量（比如：转换为英语、解析图片....）



Structure—Convolutional Blocks

- 卷积模块使用的是深度分离卷积，比传统的卷积计算更高效；
- 输入：张量[batch size, sequence length,feature channels]
- 处理：输入张量x，权重系数W（h*w大小），步幅s，放大系数d：

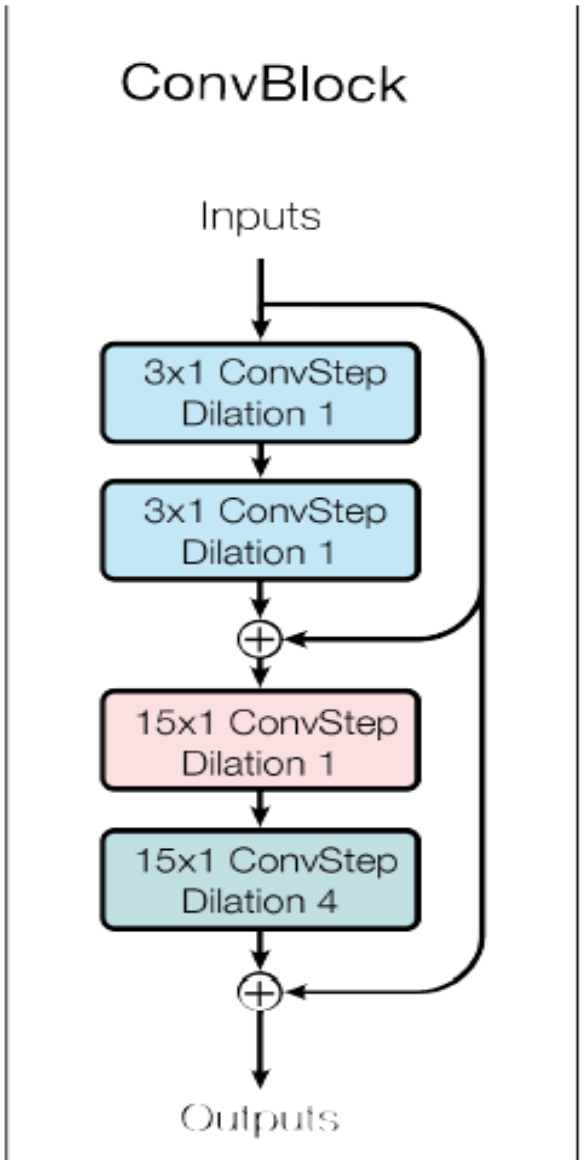
$$SepConv_{d,s,f}(W,x)$$

- 输出：同样格式的张量
- 卷积模块包含三个部分：ReLU激活函数、SepConv、正则化层

$$ConvStep_{d,s,f}(W,x) = LN(SepConv_{d,s,f}(W,ReLU(x)))$$

$$hidden1(x) = ConvStep(W_{h1}^{3\times1}, x)$$
$$hidden2(x) = x + ConvStep(W_{h2}^{3\times1}, hidden1(x))$$
$$hidden3(x) = ConvStep(W_{h3}^{15\times1}, hidden2(x))$$
$$hidden4(x) = x + ConvStep_{d=8}(W_{h4}^{15\times1}, hidden3(x))$$

$$ConvBlock(x) = \begin{cases} Dropout(hidden4(x), 0.4) & \text{during training} \\ hidden4(x) & \text{otherwise} \end{cases}$$



Structure—Attention Blocks

we use a multi-head dot-product attention mechanism inspired by [3] and similar to [1]

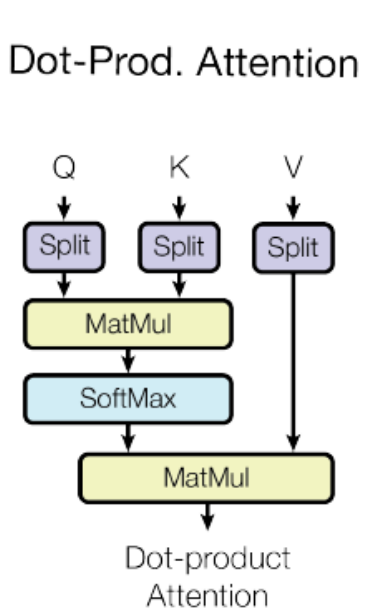
[1] Attention is all you need.

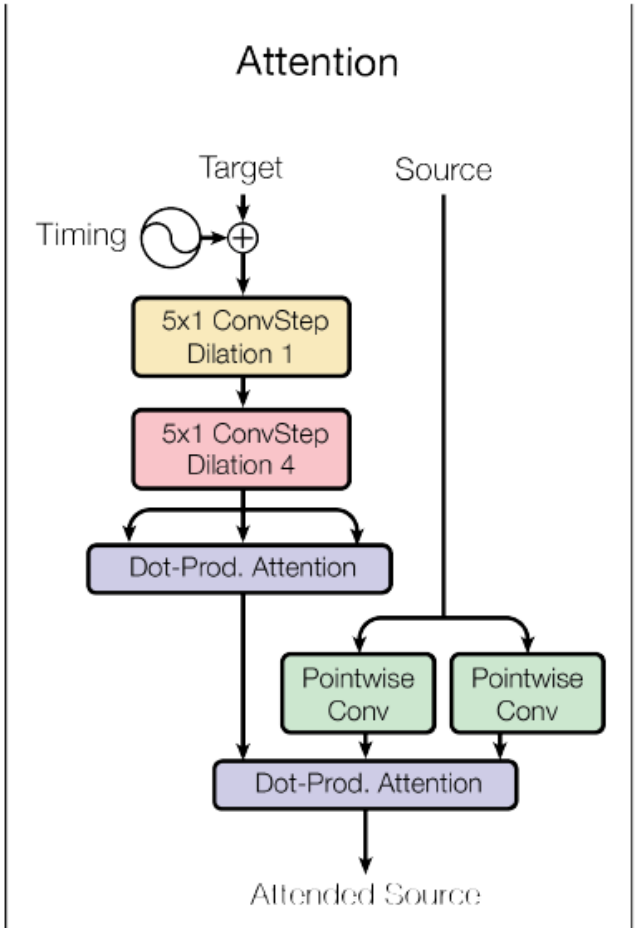
[3] Neural machine translation by jointly learning to align and translate.

Attention的两个目的：

1. 减小处理高维输入数据的计算负担，通过结构化的选取输入的子集，降低数据维度。

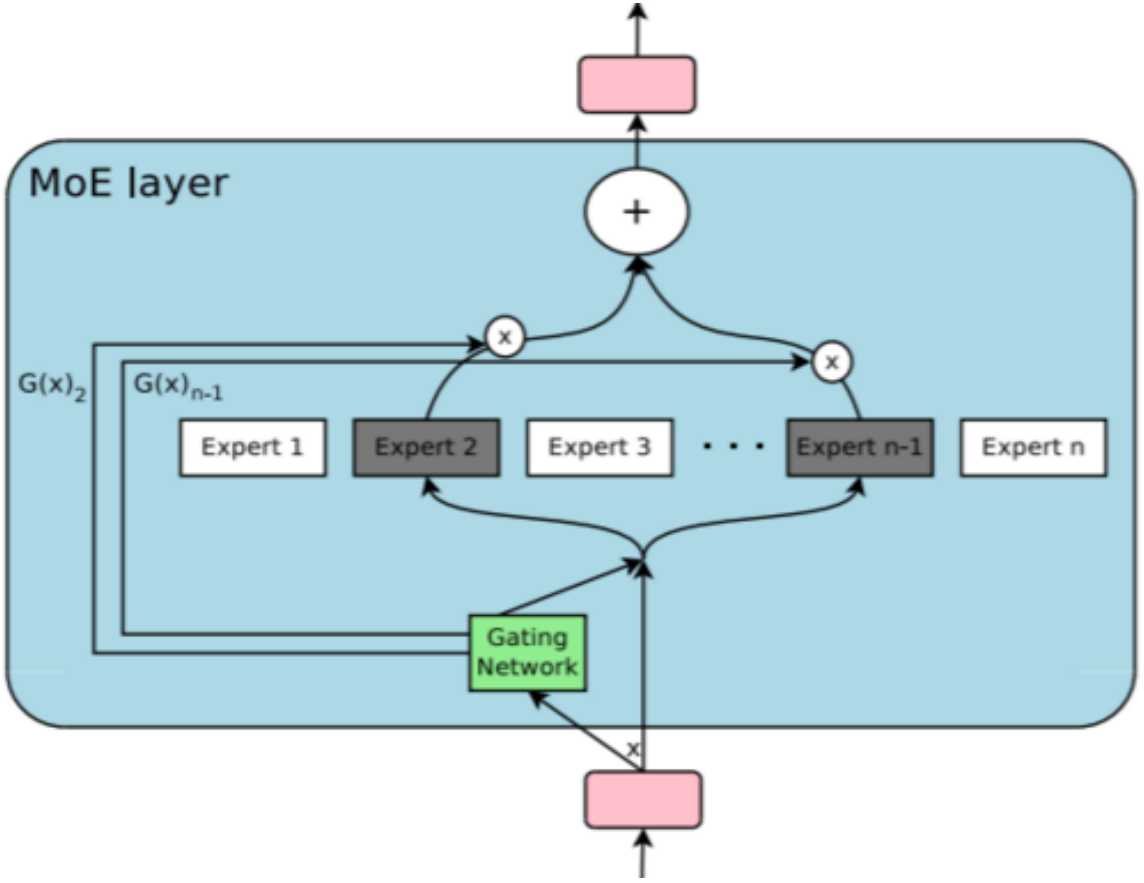
2. “去伪存真”，让任务处理系统更专注于找到输入数据中显著的与当前输出相关的有用信息，从而提高输出的质量。





Structure—Mixture-of-Experts Blocks

单一的前馈神经网络（expert）与门网络进行稀疏组合，谷歌ICLR 2017论文提出超大规模的神经网络：稀疏门控专家混合层



Structure—Modality Nets

四种模式网：语言（文本），图像，音频，分类数据。这些小型特定子网将数据转换为统一的表示形式并从中恢复。他们具有两个特点：统一的数据形式但**大小可变；同领域共用一个模式网**。

1. 语言模式网：文本数据全都通过8k的子词单元标记过；输入标记序列，映射到正确的维度，通过一系列解码映射，最终通过Softmax层输出标记词汇的概率分布；
2. 图像模式网：输入模式与Xception 输入流类似

$$
\begin{aligned}
c1(x,F) &= ConvStep_{f=F}(W^{3\times3}, x)\\
c2(x,F) &= ConvStep_{f=F}(W^{3\times3}, c1(x,F))\\
p1(x,F) &= MaxPool_2([3\times3], c2(x,F))\\
ConvRes(x,F) &= p1(x,F) + ConvStep_{s=2}(W^{1\times1}, x),
\end{aligned}
$$

$$
\begin{aligned}
h1(x) &= ConvStep_{s=2,f=32}(W^{3\times3}, x)\\
h2(x) &= ConvStep_{f=64}(W^{3\times3}, h1(x))\\
r1(x) &= ConvRes(h2(x), 128)\\
r2(x) &= ConvRes(r1(x), 256)\\
ImageModality_{\text{in}}(x) &= ConvRes(r2(x), d)
\end{aligned}
$$

3. 音频模式网：输入一维的波形或者二维的声谱图，使用8个 ConvRes 模块的叠加，每个模块 $l_i = ConvRes(l_{i-1}, 2^i)$.

Structure—Modality Nets

4. 分类模式网：与Xception输出流类似，如果输入是二维数据，输出也将转换为二维，计算如下：

$$
\begin{aligned}
skip(x) &= ConvStep_{s=2}(W^{3\times3}_{skip}, x) \\
h1(x) &= ConvStep(W^{3\times3}_{h1}, x) \\
h2(x) &= ConvStep(W^{3\times3}_{h2}, h1(x)) \\
h3(x) &= skip(x) + MaxPool_2([3\times3], h2(x)) \\
h4(x) &= ConvStep_{f=1536}(W^{3\times3}_{h4}, h3(x)) \\
h5(x) &= ConvStep_{f=2048}(W^{3\times3}, h4(x)) \\
h6(x) &= GlobalAvgPool(ReLU(h5(x))) \\
CategoricalModality_{\text{out}}(x) &= PointwiseConv(W^{classes}, h6(x))
\end{aligned}
$$

Experiments

8种训练数据，同时训练，网络结构使用TensorFlow搭建：

(1) WSJ speech corpus [7]

(2) ImageNet dataset [23]

(3) COCO image captioning dataset [14]

(4) WSJ parsing dataset [17]

(5) WMT English-German translation corpus

(6) The reverse of the above: German-English translation.

(7) WMT English-French translation corpus

(8) The reverse of the above: German-French translation.

Experiments

与最新结果的差距：

Problem	MultiModel (joint 8-problem)	State of the art
ImageNet (top-5 accuracy)	86%	95%
WMT EN → DE (BLEU)	21.2	26.0
WMT EN → FR (BLEU)	30.5	40.5

Table 1: Comparing MultiModel to state-of-the-art from [28] and [21].

单独训练与同时训练结果对比：

Problem	Joint 8-problem		Single problem	
	log(perpexity)	accuracy	log(perplexity)	accuracy
ImageNet	1.7	66%	1.6	67%
WMT EN→DE	1.4	72%	1.4	71%
WSJ speech	4.4	41%	5.7	23%
Parsing	0.15	98%	0.2	97%

Table 2: Comparison of the MultiModel trained jointly on 8 tasks and separately on each task.

Experiments

Problem	Alone			W/ ImageNet			W/ 8 Problems		
	log(ppl)	acc.	full	log(ppl)	acc.	full	log(ppl)	acc.	full
Parsing	0.20	97.1%	11.7%	0.16	97.5%	12.7%	0.15	97.9%	14.5%

Table 3: Results on training parsing alone, with ImageNet, and with 8 other tasks. We report log-perplexity, per-token accuracy, and the percentage of fully correct parse trees.

Problem	All Blocks		Without MoE		Without Attention	
	log(perpexity)	accuracy	log(perplexity)	accuracy	log(perplexity)	accuracy
ImageNet	1.6	67%	1.6	66%	1.6	67%
WMT EN→FR	1.2	76%	1.3	74%	1.4	72%

Table 4: Ablating mixture-of-experts and attention from MultiModel training.

Experiments

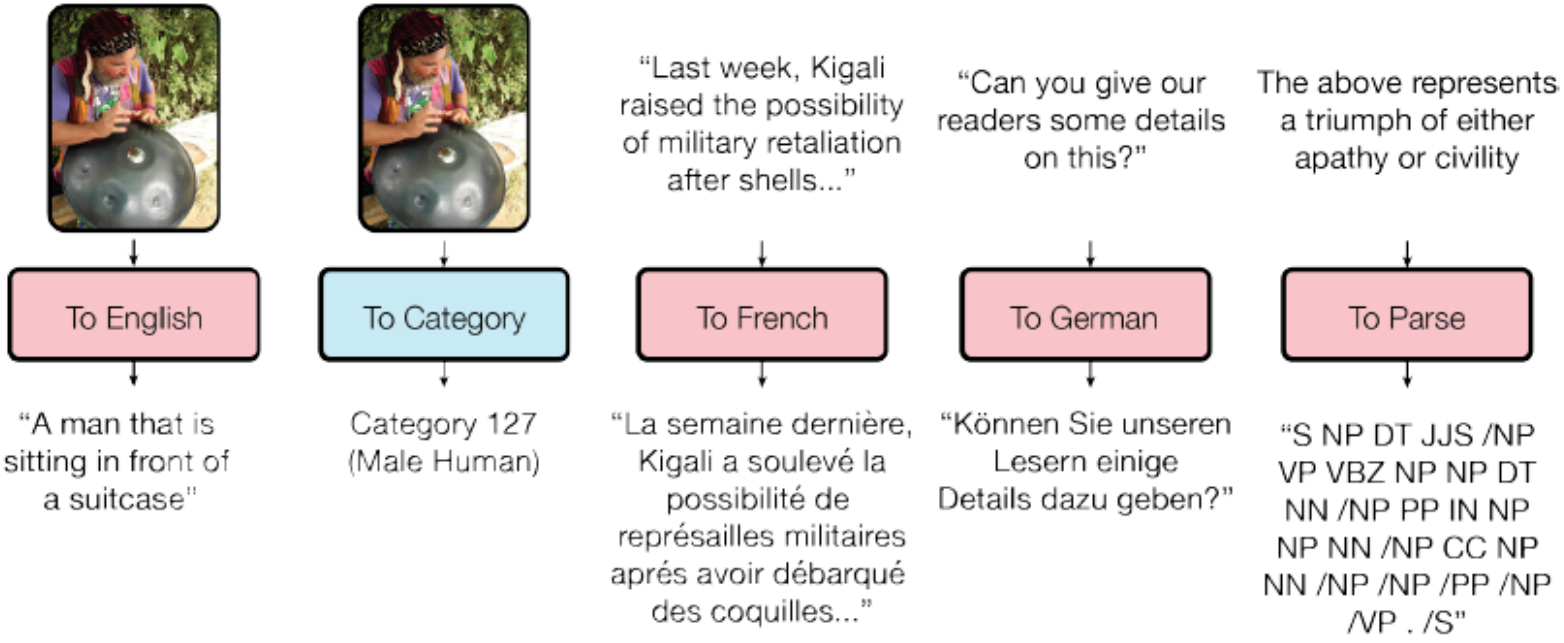


Figure 1: Examples decoded from a single MultiModel trained jointly on 8 tasks. Red depicts a language modality while blue depicts a categorical modality.

Conclusion

这种单一的深度学习模型解决多个跨领域问题是第一次被提出，模型的计算模块中大量的参数被共享，这可能是导致模型能产生良好效果的主要原因，同时为迁移学习提供很好的借鉴作用。

MultiModel架构从应用于神经机器翻译早期的编码器 - 解码器架构中抽象出来。早期使用LSTM的RNN来进行翻译，后来卷积架构在神经机器翻译中取得了很好的效果，但是这些早期的模型都是在卷积之上使用一个标准的RNN来产生输出，并有影响性能的瓶颈，全卷神经机器翻译后来又被提出，后来的ByteNet、WaveNet、Xception，MultiModel中使用了这些模型的特点。

Conclusion

缺点：

- 准确率还有一定的差距
- 训练时间，难度，权重系数大小并未提及
- 模型庞大

整体感觉：

- 文中使用的方法绝大部分都是各领域中提出的，attention、MoE、Xception、ByteNet、WaveNet；
- 对这些模型结构并不了解，读完感觉像是用别人的积木在搭建；
- 但唯一不变的核心是对不同的对象用相似的方法提取关键特征；
- 自我的基础知识架构有待提高。