07 语音与自然语言处理

斯坦福 SAIL 负责人 Christopher Manning：基于深度上下文词表征的语言结构发现

整理：智源社区 何灏宇

Christopher Manning，斯坦福人工智能实验室（SAIL）主任，斯坦福大学语言学和计算机科学系机器学习领域、斯坦福人类中心人工智能研究所（HAI）副主任。Manning 的研究目标是以智能的方式实现人类语言的处理、理解及生成，研究领域包括树形 RNN 、情感分析、基于神经网络的依存句法分析、神经机器翻译和深度语言理解等，是一位 NLP 领域的深度学习开拓者。他是国际计算机学会（ACM）、国际人工智协会（AAAI）、国际计算语言学会（ACL）等国际权威学术组织的 Fellow，曾获 ACL、EMNLP、COLING、CHI 等国际顶会最佳论文奖，著有《统计自然语言处理基础》、《信息检索导论》等自然语言处理著名教材。

Christopher Manning 的演讲主题是“Linguistic structure discovery with deep contextual word representations”，即“基于深度上下文词表征的语言结构发现”。

在演讲中，Christopher Manning 根据对语言学结构的学习程度，将语言模型分为三个发展阶段： 早期基于概率统计、无法学习语言结构的黑暗时代（Language Models in The Dark Ages)； 之后则是启蒙时代的神经语言模型（Enlightenment era neural Language Models)，特点是具备一定学习语言结构的能力；2018 年始，基于 Transformer 结构的大参数量预训练模型（Big Language Models）大行其道，Manning 发现预训练语言模型的参数中包含着非常多的语言结构信息，并在本次演讲中进行了详细的解析。

一、语言模型：用数学给语言建模

在报告中，Christopher Manning 首先引出了语言模型的概念。语言模型是对自然语言进行数学建模的工具，它提供了一种能够用数学模型去表示自然语言的方法。现如今通用的语言模型大多采用序列化概率模型的思想，比如在给定的语境下预测下一个词出现的概率。

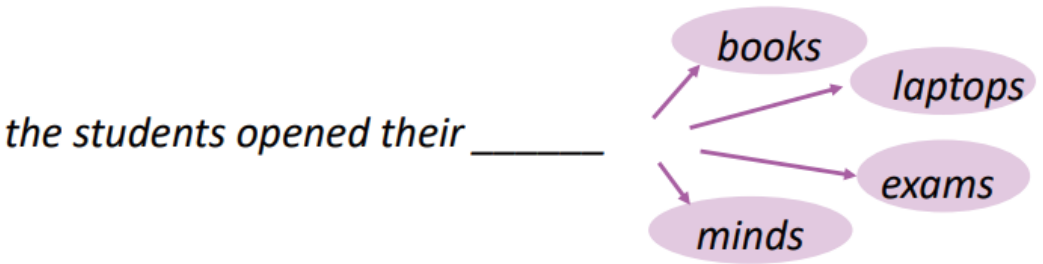


图 1： 根据语境预测下一个词

语言模型如 N-Gram 语言模型、基于循环神经网络的语言模型及预训练语言模型等都在不同的任务上被广泛使用，且能达到理想的效果。然而，这些语言模型真的学到了语言结构吗？还是说它们仅仅是在句子层面上学习词的概率分布？ Manning 给出了他的答案。

二、黑暗时代：N–Gram 语言模型

N–Gram 语言模型，是通过统计数据中给定词在长度为 n 的上文的条件下出现的频率来表征这些词在相应语境下的条件概率，如

$$P(\text{accusation}|\text{President Trump denied the}) \approx \frac{c(\text{denied the accusation})}{c(\text{denied the})}$$

图 2：N–Gram 例子

N–Gram 语言模型是神经网络出现之前构建语言模型的通用方法，该方法虽然通过引入马尔科夫假设，但是其参数量依然很大。另外，N–Gram 语言模型通过平滑和回退策略解决数据稀疏的问题。但是 N–Gram 语言模型学到了多少人类语言的结构信息？有些语言学家们认为几乎没学到。虽然这样的模型可能会包含一些简单的常识性知识，比如“船”通常会与“沉没”、“起航”等词共同出现，或者模型会学习到一些简单的词法，比如类似于“冠词 – 形容词 – 名词”这样的句子，但是 N–Gram 语言模型对于“名词”这样的词性概念和语言结构规则是没有概念的。

因此，在那个时代，如果想要让模型学习到语言结构，必须通过人工标注的方式获取特定语言结构的训练数据，然后训练相应的分类器。采用这一方法固然是能让语言模型学习到语言结构，但是标注成本太高且数据的迁移性差，似乎并不是一个好的解决方案。

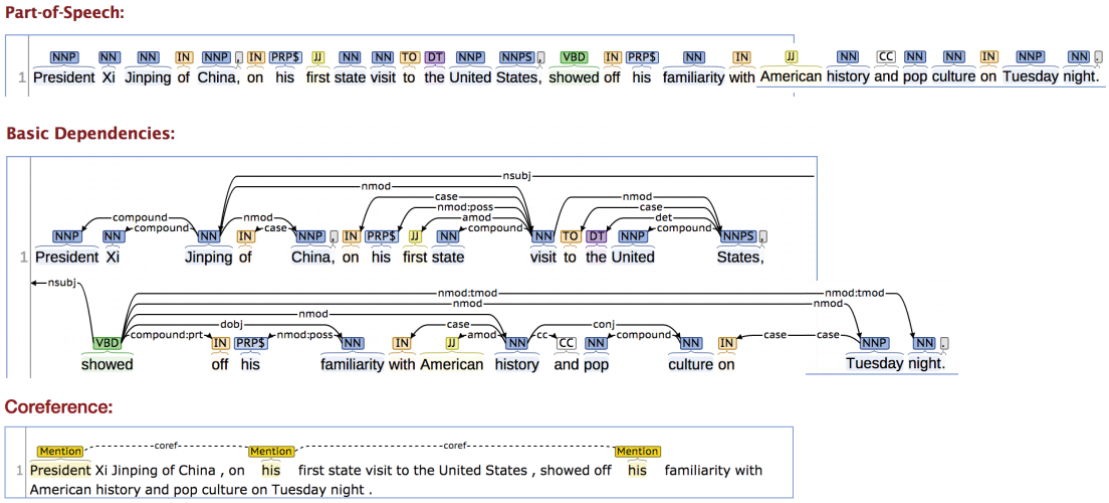


图 3：人工标注的语法

Manning 随后表示，想要让语言模型学习到自然语言的结构知识，只学习字面上的信息是远远不够的，但幸好，自 N–Gram 语言模型之后，基于神经网络的语言模型取得了长足的进步。

三、启蒙时代：神经网络赋予语言模型新的方向

得益于神经网络和深度学习带来的强大学习能力，神经网络语言模型展现出了比 N–Gram 语言模型好得多的效果，这其中最为人熟知的便是基于循环神经网络的语言模型，例如词向量模型、LSTM 模型等。词向量模型通

过把高维度的稀疏向量嵌入到低维度的分布式向量，从根本上解决了维度灾难问题。而 LSTM 则是基于循环神经网络，通过“门”的机制解决长距离依赖的问题，这样的模型结构在处理语句这种序列化数据时就有着天然的优势。Manning 提到，N-Gram 和过去的大多数模型都解决不了语句中的长距离依赖问题，但我们可以期待神经语言模型做到这一点。

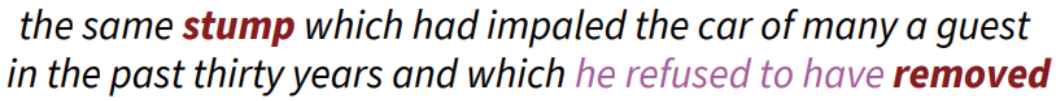


图 4：预测词 removed，需要用到句子中距离较远的词 stump 而不是通过 N-Gram 在近距离取上下文

同时，Manning 还展示了通过树结构的神经网络捕捉语句结构的一个研究成果。事实上，Manning 早期的深度学习工作一直致力于构建树形模型，因为在他看来，树形模型更能捕捉到语言不同于线性的视觉或者信号处理的结构特点。他们建立的 TreeLSTM 能够在一定程度上学习到如何去构建语句的语法树，该模型在细粒度情感分类、语义关系分类等任务上也取得了更好的效果，但比提高准确率更重要的是，语言模型终于开始学习语言结构了。

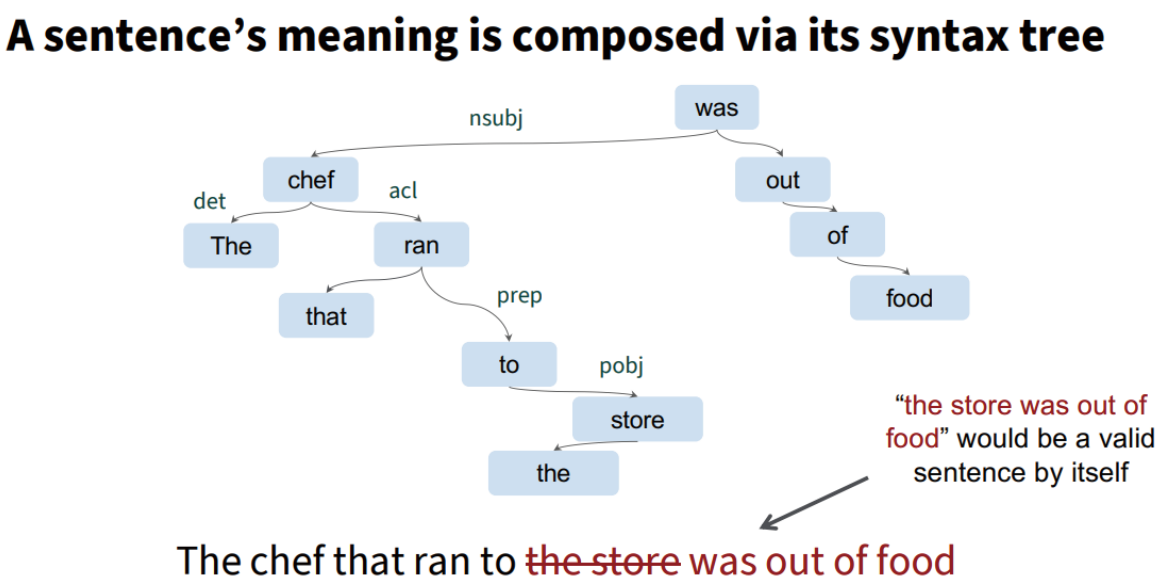


图 5：语法树

四、大模型时代：Transformer 模型带来巨大突破

2018 年，大参数量的预训练语言模型一个接一个的出现，为自然语言处理带来了突破性的进展。

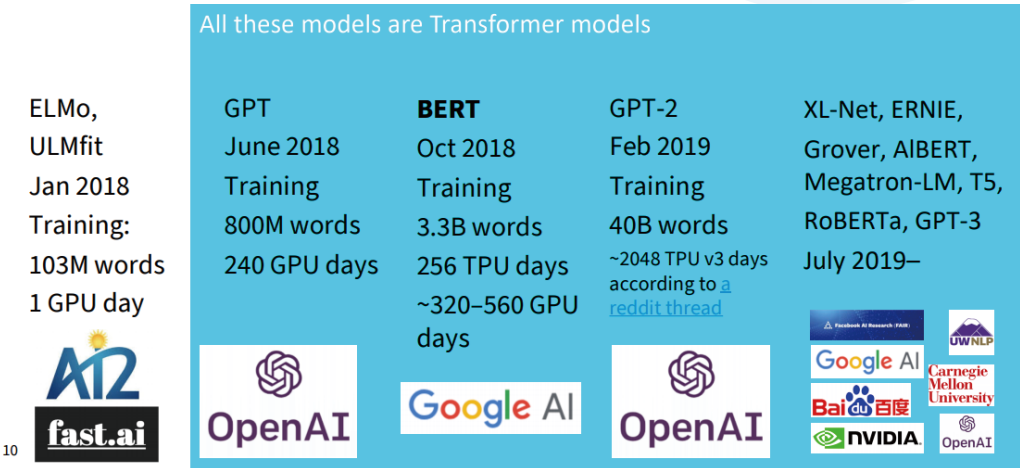


图 6：预训练语言模型

在这些预训练模型中，除了 ELMo，其他的模型都应用了 Transformer 结构，原因是 Transformer 的结构使得模型在 GPU 上进行大规模训练成为可能，而模型的参数量也越来越大，达到十亿甚至百亿级别。由于时间原因，Manning 只对 Transformer 结构进行了一个大概的描述：Transformer 的输入是句子中的词以及词的位置编码，通过一层线性变换，每个词得到 Query、Key、Value 三个低维向量。通过对三个向量做 Attention 运算，从而计算出句子中的每个词应该对句子中的其他词付出多少“注意力”。不仅如此，Transformer 结构中还引入了“多头”机制，“多头”机制认为句子中的上下文信息可以从多个方面进行挖掘，因此 Transformer 使用了多个权重矩阵对 Query、Key、Value 向量进行 Attention 运算，从而达到通过多个权重矩阵学习多重语义信息的目的。

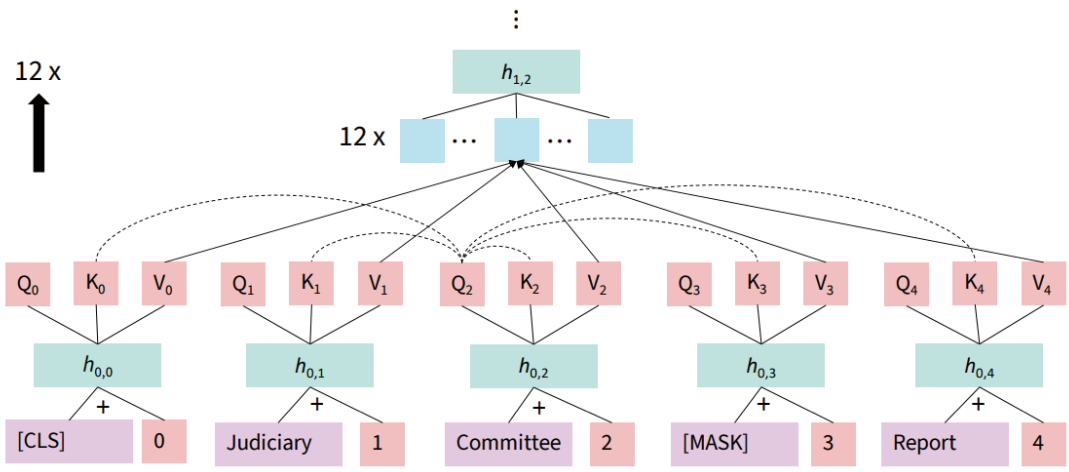


图 7：Transformer 的结构

这些基于 Transformer 结构的预训练语言模型在自然语言处理的很多领域都产生了巨大的影响，显著地提高了多个 NLP 任务的准确率。那么，动辄几十亿参数的预训练模型们可以学习到多少语言结构呢？在本次演讲中 Manning 选取了其中最著名的 BERT 模型进行了分析。

根据对 Transformer 结构的理解可以知道，Attention 运算是通过点积加权的方式计算两个向量的相关性，从而得到句子中的每个词对其他词该付出多少“注意力”。通过分析这个注意力结果，Manning 发现，在 BERT 的多个“头”中，有几个“头”是能够通过无监督或自监督的方式学习到和依存句法相关的信息的。

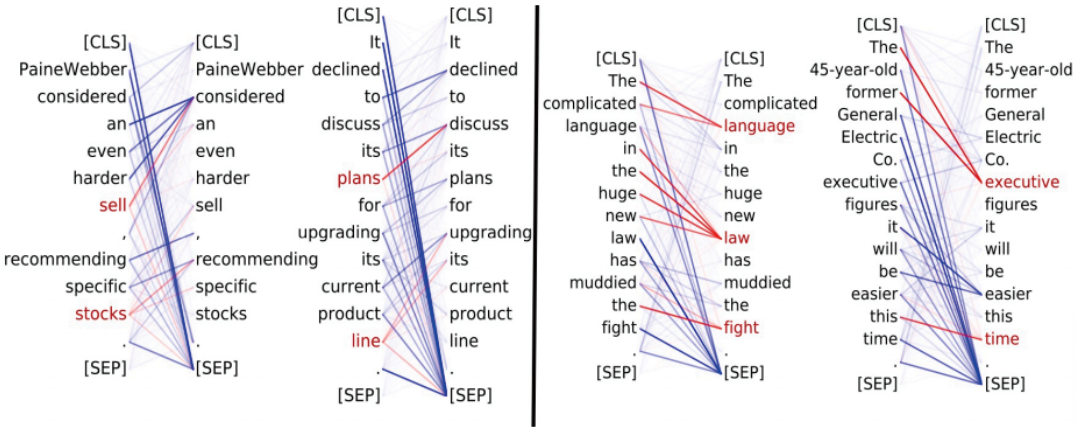


图 8：BERT 模型中每个词对其他词的注意力，颜色越深表示注意力越强

如上图的左半部分，宾语 sell 和 stocks 会将注意力更多地指向动词 considered 和 recommending，而在上图的右半部分，限定词（冠词等）the、in，形容词 huge、new 等，它们更多将注意力指向名词 language、law、flight、time 等。如果对这四个例子中的语句进行依存分析，我们会发现左图中的词 sell、stocks 与动词 considered、recommending 构成了直接宾语的依存关系，而右图中的词 the、in、huge 等都是名词 laguage、law 等的前置修饰语，它们构成了语句中的限定词依存关系。可以看到模型确实在一定程度上学习到了依存句法信息。

事实上，“多头”机制不仅学习到了句法结构，也学习到了语句中的共指关系。下图中左边的例子中，she、her、Kim 实际上指的是同一个人，从模型中的注意力分布也可以看到这种关系。右图同理。

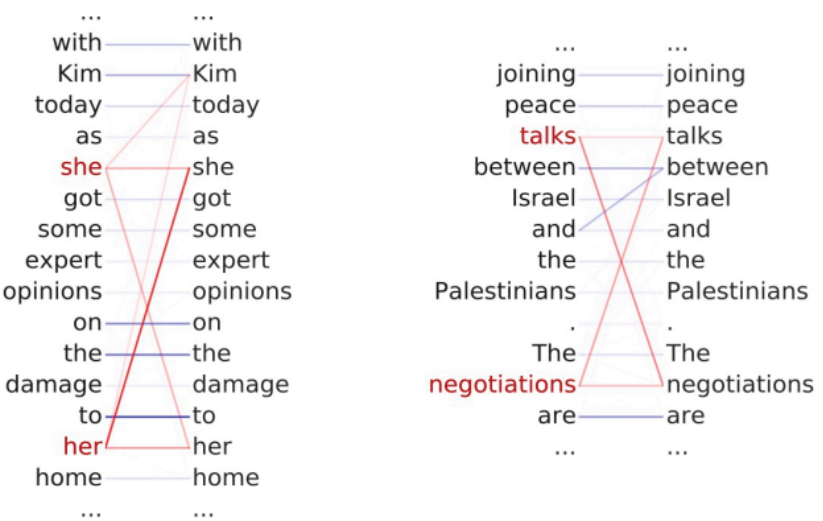


图 9：BERT 模型中的某些“头”学习到的共指关系

Manning 表示以上的这些发现证明，预训练语言模型能够对语言的符号结构进行建模，因为不管是依存句法还是共指关系其实都是一种用符号表示语法的方法，这是一件很酷的事情。但如果模型能够直接对语言结构进行建模，那就更好了。

随后，Manning 提出了一个问题： 在 BERT 模型的向量空间中是否蕴含着语法树结构？为了验证这个问题，Manning 对 BERT 模型产生的词向量进行了探索，希望这些基于深度上下文的词表征能够带给我们答案。那么，如何根据词向量去构建这些树呢?

Manning 假定句子中词向量间的 L2 距离作为树中结点之间的距离，根据这个距离构建一棵最小生成树，并将这个最小生成树作为模型学习到的语法树，最后用该树去与人工标注的语法树进行验证。值得一提的是，在不同的语境下，一个词可能会有不同的含义，那么每个词向量就可能会包含着多重语义信息。在实验时，Manning 通过对词向量进行线性变换从而将词向量映射到一个低维的空间，这个低维的向量就包含了原词向量在特定语境下的语义信息。

实验结果表明，BERT 根据上下文词表征构建的树效果非常好，在许多场景下都可以达到人工标注的精度。如下图中，根据 BERT 向量空间构建的最小生成树，与本篇文章图 5 所提到的语法树完全一致。

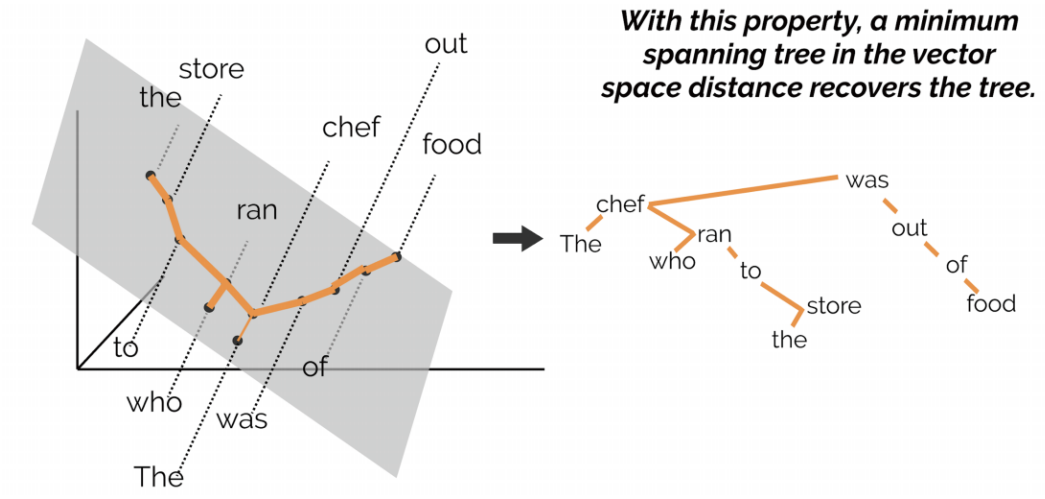


图 10：根据 BERT 向量空间构建的最小生成树

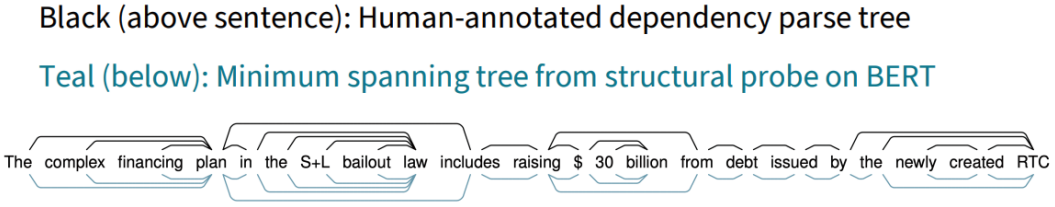


图 11：用 BERT 做语法分析树能够达到和人工标注相似的结果

根据这样的实验结果，Manning 总结道，像 BERT 这种基于深度上下文词表征的语言模型，与之前的语言模型

相比有了一个大转型，不论是形态上还是学习效果上。模型中大量的参数使得神经网络不再仅仅去学习词与词之间的表面联系，而是有了学习语法结构的能力。至于为什么模型会去主动学习语法结构，Manning 也给出了解释，他认为模型之所以会去主动学习语法结构，是因为学习语法结构能够帮助模型更好地完成预测任务，也就是说，模型本质上依然是在提高预测能力，由于学习到语法结构有助于更好地预测，模型就会利用参数去学习语句的语法结构。

接下来，Manning 做了另一个更有趣的探索，探索不同种类语言的 BERT 模型是否学到了相似的语法信息。做法如下，使用一种语言（如英语）的 BERT 模型的语法空间表示去验证另一种语言（如法语），如果验证成功，那么就说明 BERT 模型编码不同种语言的语法是采用的是相近的方法。

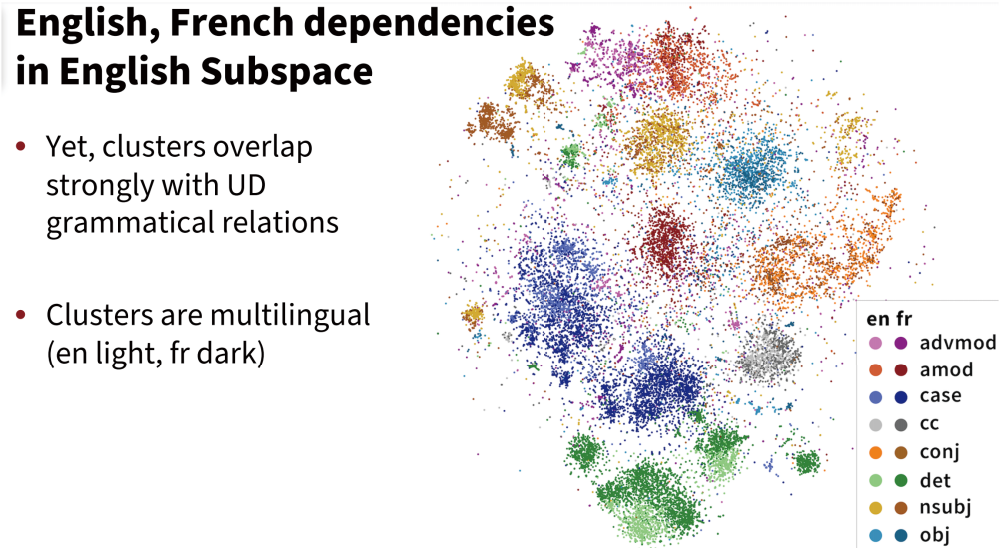


图 12：两种语言的语法空间聚类结果

上图是实验结果，相似颜色的浅色代表英文，深色代表法语，可以看到聚类效果明显，这表明 BERT 模型在建模不同语言的语法信息时采用的方法是相近的。

演讲最后，Manning 提出了如下几点思考。

1. 基于无监督或自监督学习的上下文词表征模型能够成功学习到语言结构，取得这样的成功证明了语言模型的学习实际上是一个信息量丰富的通用任务。
2. 既然语言模型已经能够学习语言结构，那过去几十年耗费人力标注语言学数据算是一个错误吗？
3. 基于深度上下文词表征的语言模型已经从之前的基于统计的关联学习模型转型，开始主动探索语言结构。
4. 在下个十年，语言模型的任务是否应该更多地将重心放在接地语言学习（Grounded Language Learning）上？

华盛顿大学教授 Mari Ostendorf：基于显式上下文表征的语言处理

整理：智源社区 何灏宇

Mari Ostendorf，于 1999 年加入华盛顿大学，电子与计算机工程系的系统设计方法学特聘教授，语言学和计算机科学与工程学的兼职教授。IEEE，ISCA 和 ACL 的院士，华盛顿州立科学院院士以及爱丁堡皇家学会会员。Ostendorf 教授目前的研究专注于探索在多方上下文的场景下理解和生成语音及文本的动态模型。

Mari Ostendorf 本次演讲的主题是《Contextualized Language Processing with Explicit Context Representation》。

在本次演讲中，Mari 介绍了她的研究成果——基于显式上下文表征的语言处理方法。该方法通过将上下文信息做显式的表征处理，从而达到在文本信息中能够有效融合上下文信息的目的。

下面是 Mari Ostendorf 本次演讲的精彩要点。

一、语言模型为何需要上下文

上下文在语言理解的过程中扮演着重要的角色，需要指出的是，这里提到的上下文并不仅仅指语句序列中的“前”和“后”，这里的上下文是指包括语境和情境在内的多种与语句相关的因素。语言对上下文的依赖表现于：演讲中的语句会依赖于演讲主题；日常交流时的语句会依赖于交流双方的社交背景，等等。人类可以自如地在不同语境中进行切换，但是计算式的语言模型在处理不同种类的上下文时性能退化会非常严重。现如今解决这一问题的方法是使用更多的数据来对模型进行微调，但 Mari 认为，我们还可以更有效地使用上下文中包含的信息。

上下文有很多形式，但大体上可以分为两大类，第一类是语境（Language Context）有关的上下文，也就是按字面意思理解的语句序列的“前”和“后”；第二类是情境（Situational Context）有关的上下文，包括语句的来源，对话的地点，对话的形式、对话的主要任务等等。在本次演讲中，Mari 主要介绍了如何通过神经序列化语言模型去表征情境有关的上下文，以及把它们与文本信息相结合的机制。

同时，Mari 将情境有关的上下文分成了两类，一类是全局上下文，通常由元数据组成，这类上下文能够覆盖一整段文本，比如说对话者的身份以及对话时间这种通用的信息，在对话过程中是不会发生变化的。还有一类是动态上下文，通常是由连续的数据组成，比如说语音或者视频。Mari 对于这两类上下文分别给出了她的表征方法。

二、全局上下文的表征方法

训练数据与实际数据的领域不匹配是一个存在已久的问题，目前解决这个问题的方法是使用更多的数据对模型进行调整。比如 RoBERTa 模型是在 160GB 的大数据量上进行训练的，但是当将该模型应用到一个全新领域的时候仍然需要使用该领域的数据进行微调。这样的方法虽然效果不错但并没有从根本上解决这个问题，且每次切换到一个新的领域都需要使用更多的数据。Mari 认为，许多时候我们其实能够知道数据属于哪个领域，但是现有的语言模型并没有有效利用这一点。比如说，当我们处理一段法庭上辩论的脚本，对于这样的脚本，我们恰巧知道每段文字是由谁说得，我们也知道说话的人是不是法官或者是律师，而且我们应该也有一些与案件相关的背景文件。这三种信息都是原数据中就有的，Mari 认为可以将这三种信息看作三个变量，通过线性变换将

这三个变量映射成能够表征上下文的向量。这一变换过程与现如今的词向量技术类似，只不过输入不再是独热编码（One Hot）而是一个由多个参数组成的元组（Tuple）。

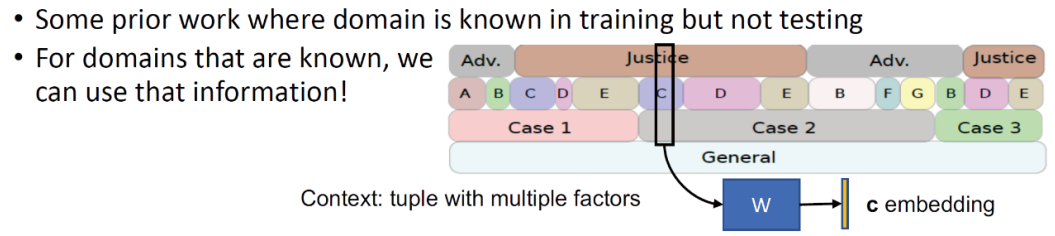


图 1：三个变量以元祖的结构进行向量化表达

一旦上下文信息被表示成向量，就可以有许多方法去将该向量融合进神经网络。比如将上下文的表征向量与词向量或者句向量拼接起来。Mari 教授在演讲中提到了她的方法，将上下文表征向量作为输入，对 RNN 的权重向量进行修正。

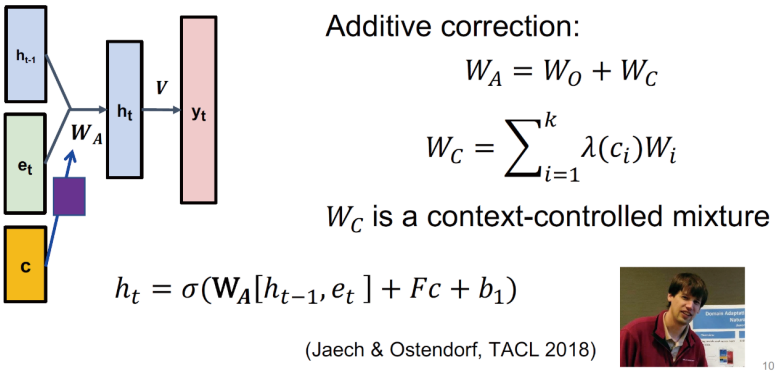


图 2：上下文表征向量 c 与权重向量进行融合

具体做法为，使用两个排序矩阵 L 和 R 与上下文表征向量 c 进行全连接运算。由于上下文表征向量 c 是一个低维稠密的向量，且 L 和 R 的维度与原本 RNN 的权重向量的维度相同，因此全连接运算后会得到一个与之前向量的维度完全相同的新的权重向量，从而达到使用上下文表征向量对权重向量进行更新的目的。由于上下文是全局的，那么 L 和 R 两个排序矩阵是具有通用性的，因此对于所有上下文向量 c，都可以通过 L 和 R 提前计算出需要被更新的那部分权重向量，这样一来，将上下文表征向量融合进权重向量的计算代价就降低了很多。Mari 称这样模型结构为 FactorCell Model。

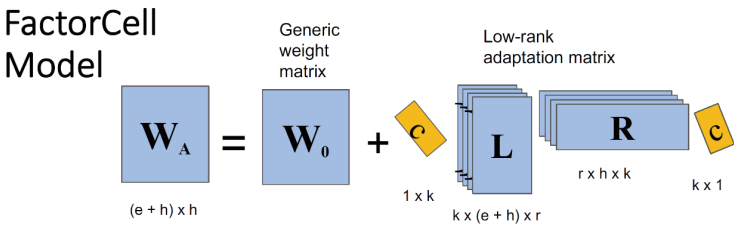


图 3：FactorCell Model 的模型结构

这种融合方式有许多优点。第一，排序矩阵 L 和 R 作为超参数会去控制上下文信息与文本信息融合的程度，当数据量发生变化时，这两个矩阵起到了缩放融合程度的作用。第二，实验发现增大排序矩阵的影响要比增大权重向量的维度更有效。第三，这个方法应用在大量不同情境下效果更好，因为它能够学习到多种上下文之间的相似程度。

Mari 教授同时还展示了该方法在 9 个数据集上进行实验的结果，她发现，对于有大量上下文的数据来说，通过该方法对 RNN 权重向量进行更新是及其有效的。传统的通过 Softmax bias 融合上下文向量的方法对于主题模型类的问题是足够的，但是对于以字为主要对象的语言模型来说并没有显著的作用。同时，实验中还发现，该方法与传统方法相比，将不同种类的上下文向量输入进模型，得到的结果在细粒度上是有差别的。

Fill in the blank: "This was my first time coming here and the food was _____".

	FC	SB
*****	amazing!	great!
****	great!	great!
***	good!	great!
**	just meh	mediocre
*	awful	mediocre

图 4：两种模型在多种语境下的结果。SB 代表传统的 Softmax bias 方法，FC 代表该演讲中的方法 FactorCell Model，* 代表不同语境

三、从多模态角度对动态上下文进行表征

能够采用前文所述方法对上下文信息进行融合的前提是，我们可以在对话开始之前就了解全局的上下文信息，从而做到能够以一种显式的方式对上下文信息进行表征。然而，在处理动态场景（如语音信号）时，我们并不能够提前知道这段对话的上下文信息，比如在语音对话开始前，我们没法知道每段话是由哪个人说的，也没法知道说话的人的情绪和意图是什么。在这种情况下，之前提到的处理全局上下文信息的方法是无法奏效的。

Mari 教授向我们介绍了一种全新的方法，她认为语音中的韵律和语调都携带着有用的信息，比如韵律和语调可以用来划分句子，在英文对话中，韵律和语调还能够体现出讲话人的意图和情感。在现实生活中，人类在听一段语音时会通过讲话者语调的高低判断这个人的情绪以及讲话人的目的，但是传统的语音转文字技术却并没有有效利用这部分信息。没有有效利用这部分信息的根本原因是，像语调这种情境信息没法显式地表示出来，不同的人在不同的背景下说不同的话时，语调都会有很大的差别。如果想要将语调这种上下文信息与文本信息相融合，首先需要解决怎样显式地表征这种上下文信息。需要了解的是，这是一个多模态问题，因为语调来源于语音信号，而文本信息来源于文字。

Mari 教授将这一多模态问题与图像描述（Image Caption）这一多模态问题进行类比，为我们展现了她在处理这

一问题时的思路。图像描述问题的主要任务是对特定的图片进行文字描述，在这个问题中，文字和图片构成了多模态，二者包含的信息互有补充，同时，文字和图片中也包含了许多冗余的信息，这与语音问题中语调和文字的特点相类似。文字描述任务中，为了解决冗余信息，通常会规定文字描述依赖于一些图片中的边框，也就是说文字只去描述那些图片中被边框框起来的部分。对应到语音问题中，Mari 将语调中的重音或者断点看作边框，这些部分是讲话者在语调中最想要表达出来的信息。

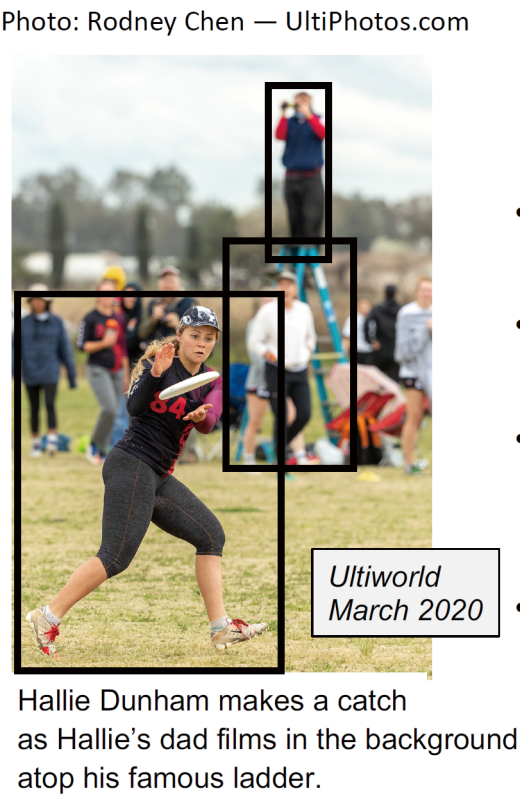


图 5：图像描述问题中，文字描述会专注于图片中被边框框起来的部分

想要解决多模态问题还需要思考的一个问题是怎样制定学习策略，也就是说多个模态之间如何通过学习紧密结合在一起，Mari 教授提出了一种归一化的方法。

具体做法是，给模型提供一段文本，模型会先去预测这段文本的语调在这段话中的分布情况，然后模型会对比预测值与观测到的实际语调之间的归一化差分指数 Z，这个指数就用来作为语调特征。最终得到的语调特征是一个低维稠密向量，方便进行深度学习的计算。根据语调特征的定义，我们可以发现它表示的是文本信息中并不包含但却存在于语音信号中的那部分信息，也就是动态上下文中包含的情境信息，Mari 称部分信息为 Innovations。

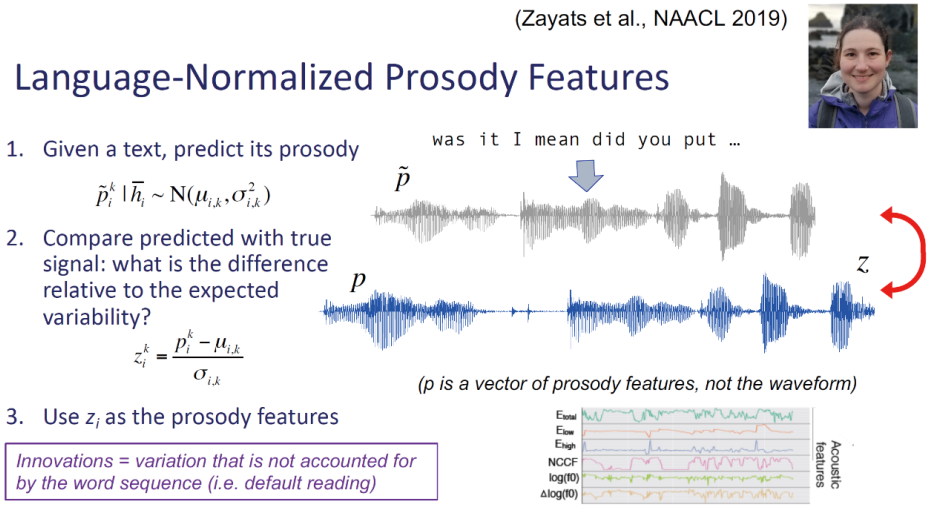


图 6：多模态上下文信息的表征方法

基于以上的的方法，在文本顺滑（Disfluency Detection）任务上做了实验，实验结果表明，原始的未经处理的语调信息用于文本顺滑任务效果并不好，但使用归一化差分后的语调特征却能够达到与纯文本相近的效果。Mari 也表示，如果将这部分语调信息与文本信息结合起来会有更多的应用价值。

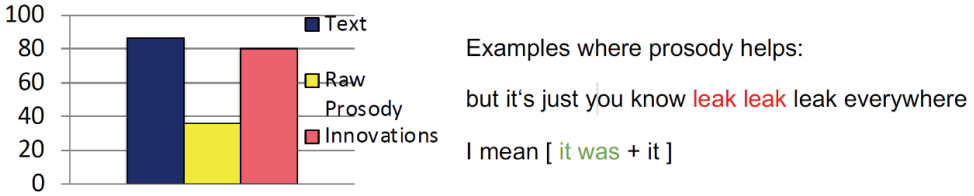


图 7：文本顺滑任务的实验结果

四、总结

通过本次演讲，Mari Ostendorf 为我们清晰展现了对上下文进行显式表征的方法，本文做了如下总结：

1. 语境信息不匹配的问题在自然语言处理的很多应用场景都有着不同程度的负面影响，然而在很多场景下，比如本次演讲中提到的全局上下文，我们是可以提前对上下文信息进行显示表征的。因此语境信息不匹配的问题可以通过对上下文信息进行向量表示，再有效地将上下文信息与文本信息进行融合来解决。
2. 除了我们熟知的词向量，其他种类的上下文信息（语境、情境等）也可以通过同样的方式被表示成低维稠密的向量。
3. 常用的使用上下文信息的方法是将表示上下文的向量与词向量或者句向量进行拼接，其实还可以有其他的更有效的方法，如本次演讲中提到的多模态融合方法。
4. 还有更多种类的上下文信息和语言模型结构等待研究者们去进行探索。

微软亚洲研究院副院长周明：预训练模型在多语言和多模态任务中的应用

整理：智源社区 徐武涛

微软亚洲研究院副院长周明本次的报告主题为《预训练模型在多语言和多模态任务中的应用》。在报告中，周明详细阐述了预训练模型的原理，以及预训练在多语言、多模态等任务中的应用。在报告最后，周明强调，“预训练模型会带来新的研究机会”。

周明，微软亚洲研究院副院长、国际计算语言学会前会长、中国计算机学会副理事长、中国中文信息学会常务理事，中国多所大学的博士生导师。他也是首都劳动奖章获得者。在微软亚洲研究院，周明博士长期领导自然语言处理方面的研究工作，主编了《机器翻译》,《智能问答》等技术专著，并且筹划组织了 NLPCC、语言与智能高峰论坛等学术会议，主导创建了 ACL 的亚洲分部。

以下是智源社区编辑对周明演讲的要点整理。

一、预训练模型的基本原理

最近 NLP 领域最重要的进展就是预训练模型了。它通过自监督学习机制从语料库学习与任务正交的知识，然后再用某一具体任务的标注数据对神经网络进行微调。具体而言，包括如下几个步骤：

(1) 预训练步骤，获得与具体任务正交（就是无关）的预训练模型；
(2) 第二个步骤是微调，针对具体任务修正网络。

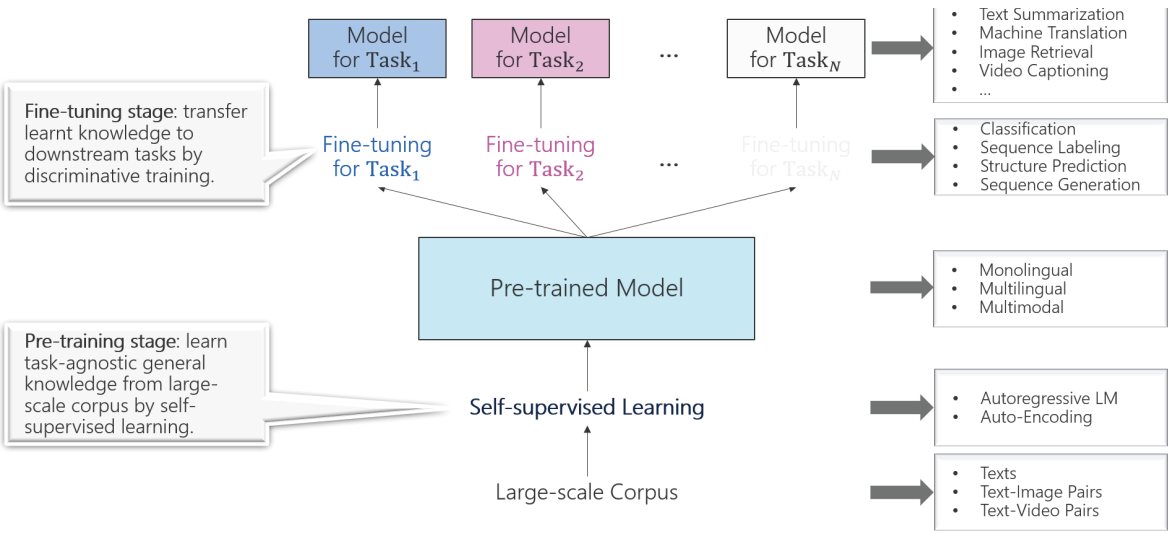


图 1：预训练模型，NLP 的新范式

训练数据可以是文本、文本 – 图像对、文本 – 视频对。预训练模型的训练任务可以使用自监督学习技术（如自回归语言模型和自动编码技术），可以训练单语言、多语言和多模态的模型。此类模型经过微调之后，可用于支持分类、序列标记、结构预测和序列生成等各项技术，也可以支持文摘、机器翻译、图片检索、视频注释等应用。

为什么我们要做预训练模型呢?

首先，预训练模型是一种迁移学习的应用，利用几乎无限的文本，学习输入序列的每一个成员的上下文相关的表示以及整体的输入序列的表示，它隐式嵌入了一般语言知识。

第二，它可以将从开放领域学到的知识转移到下游任务，以增强低资源任务和低资源语言任务。

第三，预训练模型在几乎所有 NLP 任务中都取得了目前最佳的成果。

最后，这个预训练模型 + 微调机制具备很好的可扩展性，在支持一个新任务的时候，只需要利用该任务的标注数据进行微调即可，无需对任务本身的领域知识进行调整。

下面介绍预训练模型的几个关键技术。

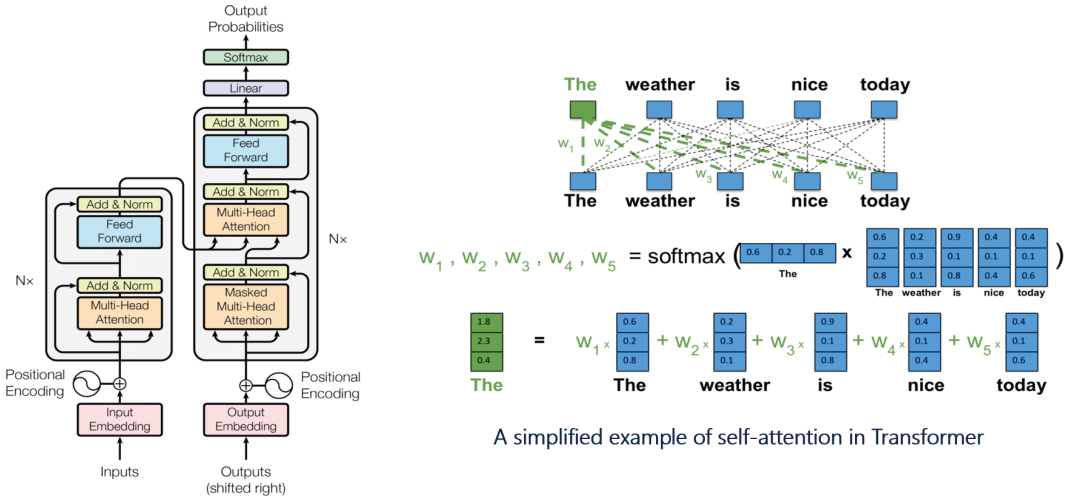
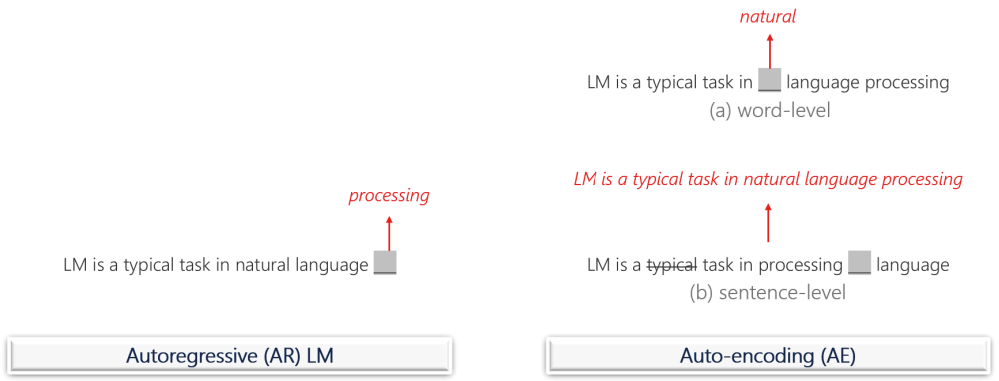


图 2：以 Transformer 作为主干

首先是 Transformer 技术。它是目前所有预训练模型的基础技术。对每一个输入序列的 Token，引入了自注意机制得到其上下文相关的表示，同时利用多头机制得到多个侧面的特征表示。如图 2 左图所示。输入层每一个 Token 的词向量和其位置向量叠加后，通过自注意模型计算与其他 Token 的相似度，把所有 Token 的词向量加权平均得到本单词的动态表示，再经过残差网络引入上一层信息来增强输入。通过前馈神经网络，经过一个非线性变化得到输出隐状态向量。这个过程可经多层变化。这样每一层有多个隐变量，每一个隐变量对应一个输入序列的 Token。在最后一层就可得到每一个 Token 的编码。然后在解码输出的时候，当前状态的隐状态，与编码层的隐状态进行自注意力计算，然后再经过非线性输出一个隐状态的向量，解码也可以经过多层。最后一层经过 Softmax 得到每一个词的输出概率。

右边显示了如何来计算自注意向量中的权值，每一个权值都是当前这个词跟其他词计算的相似度得到的全值向量，每一个词新的多维向量表示，就是所有跟它连接的所有词的加权求和表示。



Self-supervised learning is a form of unsupervised learning where the data itself provides the supervision.

图 3：通过自监督学习的预训练

第二个关键技术是自监督学习。自监督学习是一种无监督学习的形式，数据本身提供监督。在预训练的模型中，AR（自回归）语言模型 和 AE（自动编码器）是最常用的自监督学习任务，其中，自回归语言模型旨在通过自回归模型估计文本的概率分布（就是语言模型，前面的词序列对写一个词的预测概率）。自动编码器旨在从损坏的输入，比如遮掩了句子某个词、或者打乱了词序等等，重建原始数据。通过这些自监督手段来学习单词的上下文相关表示。

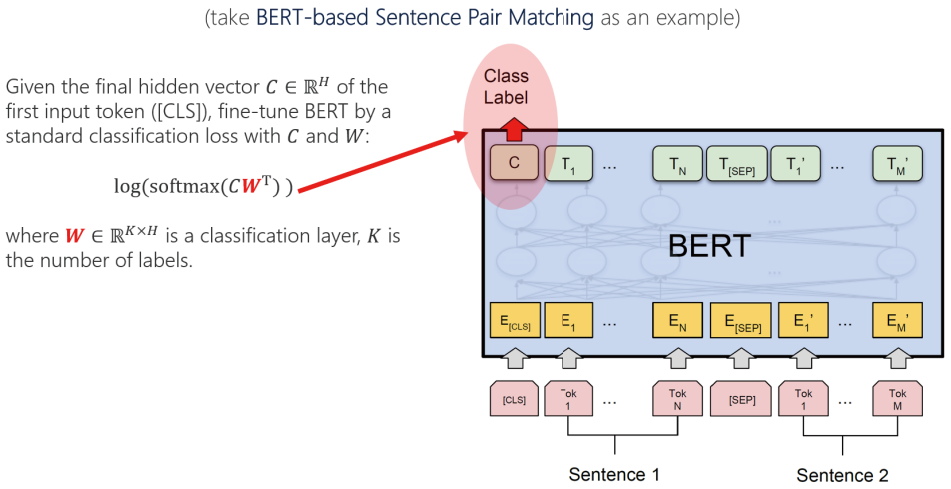


图 4：通过区分性训练进行微调

在具体任务时候，微调旨在利用其标注样本对网络的参数进行调整。举例来讲，我们使用基于 BERT 判断两个句子是否语义相同。输入是两个句子，经过 BERT 的预训练，可以得到每个词所对应的隐状态，表征每一个词的语义。我们可以简单地用预训练模型的第一个隐节点 [CLS] 预测分类标记（C）。预测损失可以反传给 BERT 再整体进行微调。

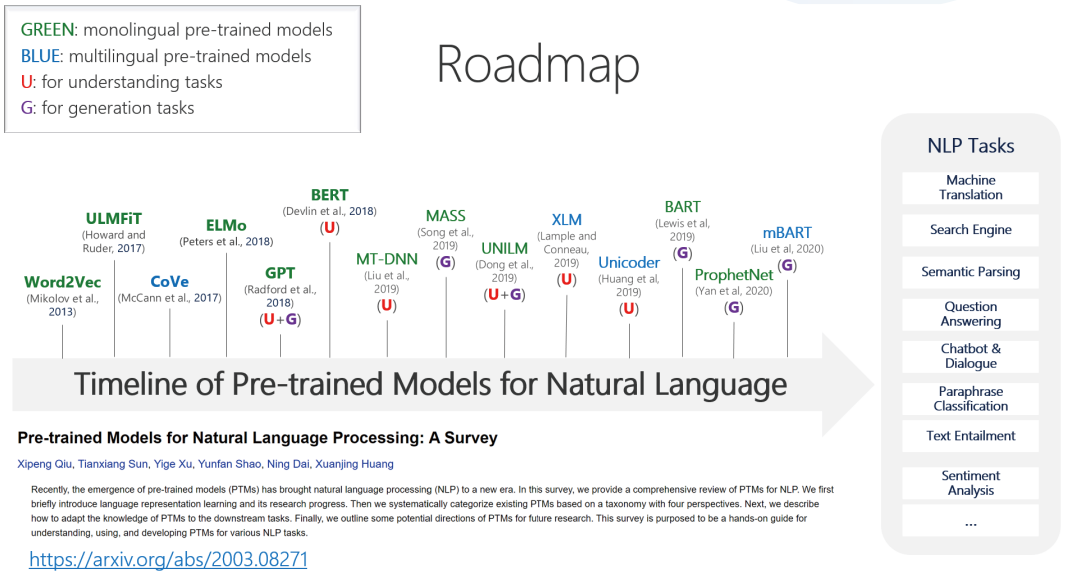


图 5：自然语言的预训练模型发展路线图

这是预训练模型的发展路线图，我列出了最有代表性的模型，这里简要介绍一下。

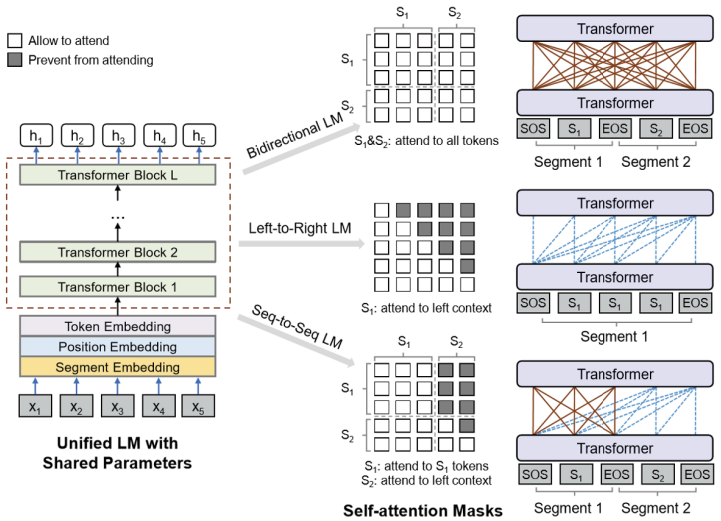
- Word2Vec，第一个使用大规模语料库计算词嵌入的模型，训练的结果是词的嵌入，是一个静态表示。
- ULMFiT，是第一个使用 RNN 基于语言模型训练的上下文相关的预训练模型。
- CoVe，利用翻译任务来训练编码器 – 解码器，并使用编码器作为预训练模型。
- ELMo，使用双向 LSTM 合并两个方向的隐状态获得上下文相关表示。
- GPT，采用语言模型进行训练，它是基于 Transformer 的单向预训练模型。
- BERT，是基于 Transformer 的基于掩码的预训练模型。
- MT-DNN，基于 BERT 增加了一些任务进行多任务训练。
- MASS，使用编码 – 解码器来训练预训练模型。
- UNILM，尝试同时支持语言理解和生成任务。
- XLM，是一种支持多语言的 BERT 模型。
- Unicoder，引入若干新的任务改进了 XLM。
- BART，是一种编码 – 解码器模型，通过还原损坏的句子训练。
- mBART，将 BART 理念扩展到多语言。

这里给大家介绍一下预训练模型的基本的过程，以 BERT 为例。BERT 是基于 Transformer 的预训练模型。它的基本模型（BERT Base）为 12 层的 Transformer；当然还有一个大型模型（BERT Large）为 24 层模型。这里说明一下几个关键地方：

1. 针对一个数据集合，BPE 工具自动获得该数据集的 Token 的集合，取频率最高的前 N 个 Token 作为词表。其他的 Token 都看作是 UNK（Unknown Word）；
2. 对数据集合的每一个数据，通过 BPE 做 Tokenize，形成 Token 的序列；
3. 训练时候，每一个 Token 有一个多维向量表示，比如 1024 维。随机初始化；

4. 计算预测的损失，然后反向传播来调整各层的网络参数。也包括每一个 Token 的多维向量表示；
5. 最后训练的结果包括：每一个 Token 的多维向量表示，每一层的网络参数，各个 Attention Model 的参数等。
6. 在用预训练模型的时候，把输入序列 Tokenize 之后，对每一个 Token，从词典中得到多维向量表示。然后根据每一层的网络参数，计算输出。

BERT 的训练任务有两个，第一个使用了掩码语言模型，就是盖住某一个词，让网络猜这个词是什么。这里当然有一些猜错，猜错的时候，计算它的损失，利用损失来调整网络。BERT 还有一个训练任务，叫 NSP，对下一句的预测，旨在预测这两个句子中组成的对，第二个句子是否在原文中是第一个句子的下一句，预测的损失也回传网络，调整网络参数，最后得到预训练的结果。



	ROUGE-1	ROUGE-2	ROUGE-L
Extractive Summarization			
LEAD-3	40.42	17.62	36.67
Best Extractive	43.25	20.24	39.63
Abstractive Summarization			
PGNet	39.53	17.28	37.98
Bottom-Up	41.22	18.68	38.34
S2S-ELMo	41.56	18.94	38.47
UNILM	**43.47**	**20.30**	**40.63**

Table 3: Evaluation results on CNN/DailyMail. Models in the first block are extractive systems listed here for reference, while the others are abstractive models. The results of the best reported extractive model are taken from (Liu, 2019).

	EM	F1
RMR+ELMo (Hu et al., 2018)	71.4	73.7
$\text{BERT}_{\text{LARGE}}$	78.9	81.8
UNILM	**80.5**	**83.4**

Table 4: Extractive question answering results on the SQuAD development set.

图 6：UniLM 模型 (Dong et al.2019)

UniLM 是由我们小组（微软亚洲研究院）开发的，是一种先进的预训练模型，用于语言理解和生成任务。首先它是一个 Transformer 的机制，用了三个任务训练：

(1) 掩码语言模型（就是自编码）类似于 BERT，利用左右词汇预测被盖住的词；
(2) 自回归语言模型，类似 GPT，利用已有的词序列预测下一个词；
(3) 编码 – 解码模型，利用输入句子和已经输出的词来预测接下来的词。

这三个任务进行多任务训练。通过一个掩码矩阵控制哪些词可以进行 Attention。训练得到的模型具备了理解和生成两种能力。在 GLUE 任务集合、文摘生成和答案抽取等任务上都取得了当时最好的水平。预训练模型已广泛应用于产品，比如提高搜索的相关性等； 也可以用于问题生成，给定一个文本，生成关于这个文本的若干问题。

二、预训练模型在多语言任务中的应用

有的时候，我们要训练多语言的任务，比如有 N 种语言，A、B、C、……、X，每一种语言都可能有自己的单语言库，有时候双语之间也存在一些并行的语料库，我们把它加在一起，来训练一个共同的语言模型——一个

跨语言的预训练模型。

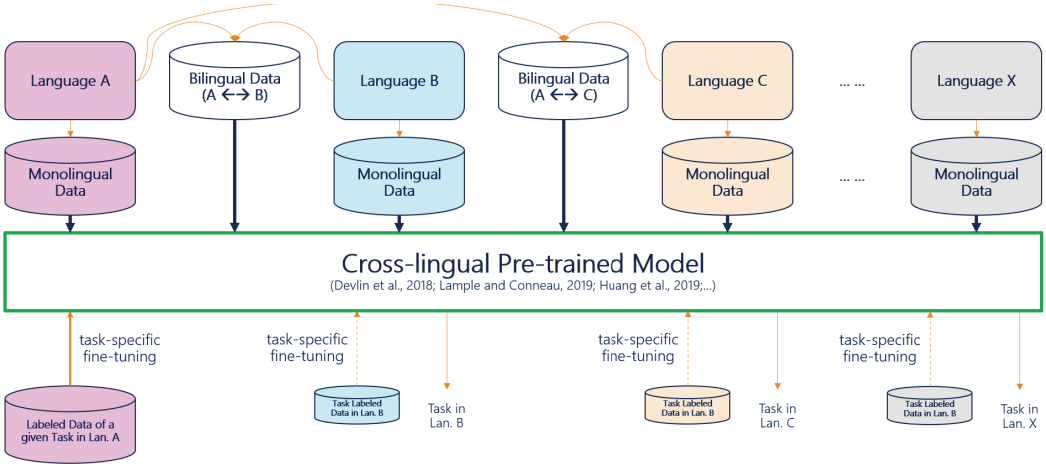


图 7：跨语言预训练

其中来自两种不同语言的单词，如果具有相似的含义，将联系在一起。然后，对某些语言，如果有带标注的数据，我们便可以在与训练模型的基础上进行微调，得到的模型应用于其他语言也能起到了一定效果。如果其它语言也有自己的标注数据的话，还可以进行进一步微调，使效果得到进一步的提升。

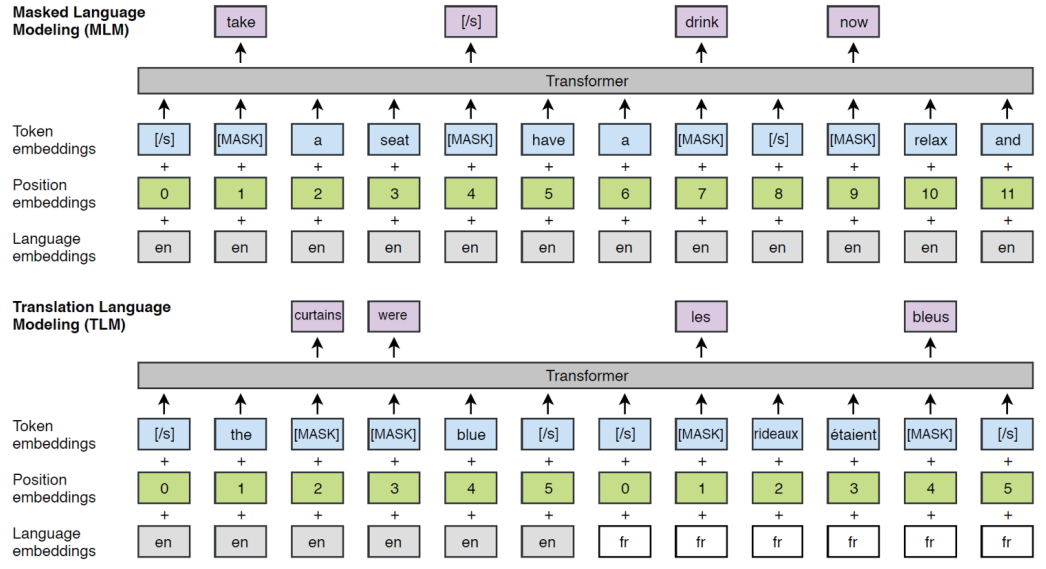
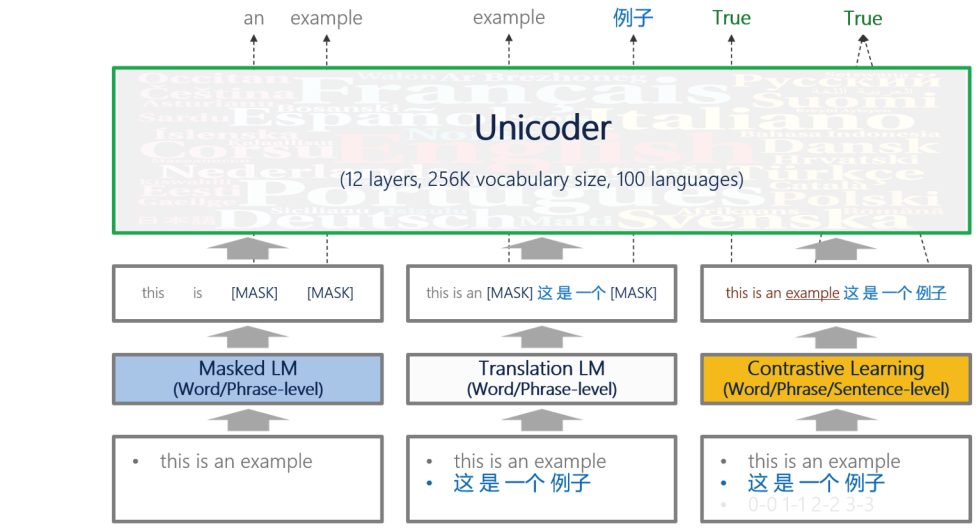


图 8：XLM 模型（Lample and Conneau,2019; Conneau et al.,2019）

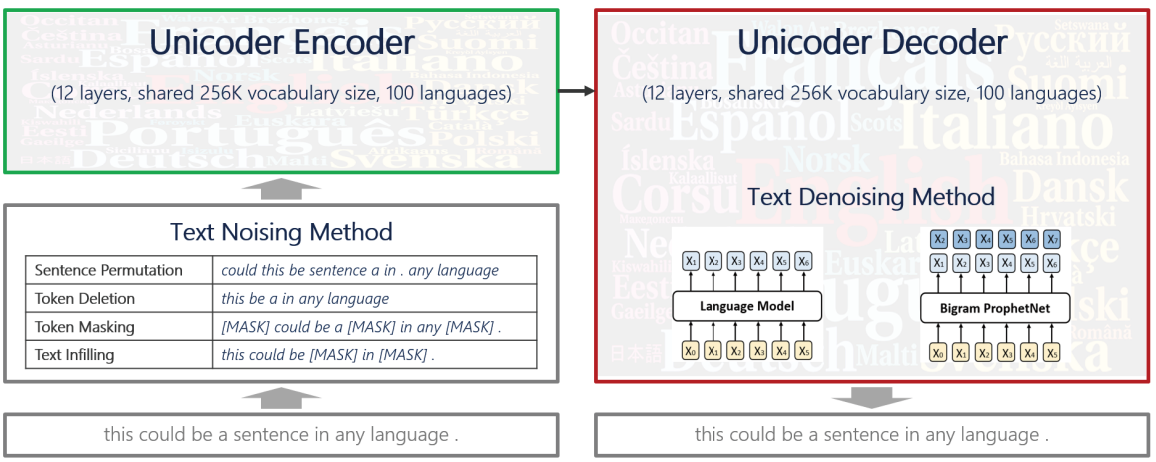
XLM 是把 BERT 扩展到多语言理解任务的一个预训练模型。XLM 中使用了两个任务。第一个是掩码（屏蔽）语言模型，与 BERT 中类似，但输入是来自多种语言的句子。通过共享所有语言的模型参数和词汇，XLM 可以获得跨语言功能。第二个任务是 TLM（翻译语言模型），它叫做翻译，但却并没有考虑对译关系。可以认为就是多语言句子对照句对，看作一个语言，去训练掩码语言模型。



Haoyang Huang, Yaobo Liang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, Ming Zhou. **Unicoder: A Universal Language Encoder by Pre-training with Multiple Cross-lingual Tasks**. EMNLP, 2019.

图 9：Unicoder 模型（Huang et al.,2019）

我们在此基础上，开发了一个叫做 Unicoder 的预训练语言模型，增加了跨语言训练新任务。除了在单语句子上进行单词和短语层面的“掩码语言模型”，以及对双语句子进行掩码语言模型（称作翻译语言模型）之外，我们增加一个新的训练任务：在利用了 Giza+ 做了单词对齐之后，通过预测两个单词的对译关系是否存在，或两个句子是否是一个互译的句子，进行网络训练，进一步提升预训练的效果。这个任务可以在单词级别做、短语级别做，也可以在句子级别做。不仅仅正例，也引入了反例，通过对比式学习，加强学习效果。



Yaobo Liang, Nan Duan, Yeyun Gong and Others. **XGLUE: A New Benchmark Dataset for Cross-lingual Pre-training, Understanding and Generation**. arXiv, 2020.

图 10：Unicoder–2 模型（Liang et al.,2020）

最近，我们还将 Unicoder 扩展到跨语言生成任务中。这个预训练模型，Unicoder–2，做了如下任务训练：给定来自多语言语料库的输入句子，我们首先打乱其句子，通过文本加噪音，然后通过解码器尝试恢复。解码时候可以用传统方法每次仅仅预测一个 Token，也可以通过我们最近的 Prophet（先知）网络预测两个 Token 或多个 Token，然后取第一个词输出，再接着预测下一个位置的 Token。这样做预测能力有新的提高。

Tasks in XGLUE

Task	# of Languages	$\|Train\|^{en}$	$\|Dev\|^{avg}$	$\|Test\|^{avg}$	Metric	Data Source
NER	4	15.0K	2.8K	3.4K	F1	ECI Multilingual Text Corpus
POS	18	25.4K	1.0K	0.9K	ACC	UD Tree-banks (v2.5)
NC*	5	100K	10K	10K	ACC	Commercial News Website
MLQA	7	87.6K	0.6K	5.7K	F1	Wikipedia
XNLI	15	433K	2.5K	5K	ACC	MultiNLI Corpus
PAWS-X	4	49.4K	2K	2K	ACC	Wikipedia
QADSM*	3	100K	10K	10K	ACC	Commercial Search Engine
WPR*	7	100K	10K	10K	nDCG	Commercial Search Engine
QAM*	3	100K	10K	10K	ACC	Commercial Search Engine
QG*	6	100K	10K	10K	BLEU-4	Commercial Search Engine
NTG*	5	300K	10K	10K	BLEU-4	Commercial News Website

Table 2: 11 downstream tasks in XGLUE. For each task, training set is only available in English. $|Train|^{en}$ denotes the number of labeled instances in the training set. $|Dev|^{avg}$ and $|Test|^{avg}$ denote the average numbers of labeled instances in the dev sets and test sets, respectively. $*$ denotes the corresponding dataset is constructed by this paper.

Task	ar	bg	de	el	en	es	fr	hi	it	nl	pl	pt	ru	sw	th	tr	ur	vi	zh
NER			✓		✓	✓				✓									
POS	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓
NC*			✓		✓	✓	✓						✓						
MLQA	✓		✓		✓	✓		✓										✓	✓
XNLI	✓	✓	✓	✓	✓	✓	✓	✓					✓	✓	✓	✓	✓	✓	✓
PAWS-X			✓		✓	✓	✓												
QADSM*			✓		✓		✓												
WPR*			✓		✓	✓	✓		✓			✓							✓
QAM*			✓		✓		✓												
QG*			✓		✓	✓	✓		✓			✓							
NTG*			✓		✓	✓	✓						✓						

Table 3: The 19 languages covered by the 11 downstream tasks: *Arabic* (ar), *Bulgarian* (bg), *German* (de), *Greek* (el), *English* (en), *Spanish* (es), *French* (fr), *Hindi* (hi), *Italian* (it), *Dutch* (nl), *Polish* (pl), *Portuguese* (pt), *Russian* (ru), *Swahili* (sw), *Thai* (th), *Turkish* (tr), *Urdu* (ur), *Vietnamese* (vi), and *Chinese* (zh). All these 6 new tasks with $*$ are labeled by human, except es, it and pt datasets in QG (80+% accuracy) are obtained by an in-house QA ranker.

NER: Named Entity Recognition
POS: Part-of-Speech Tagging
NC: News Classification
MLQA: Multilingual MRC
XNLI: Natural Language Inference
PAWS-X: Paraphrase Classification
QADSM: Query-Ads Matching
WPR: Web Page Ranking
QAM: Question-Answer Matching
QG: Question Generation
NTG: News Title Generation

图 11：多语言任务评测数据集 XGLUE

我们建立了多语言任务的评测数据集 XGLUE。上图是 XLGUE 中的 11 个下游任务。其中包括 NER、POS 等等。现在 XGLUE 已经发布。这些任务现在涵盖 19 种语言。大家可以登录 XGLUE 的 Leader board，自己提交任务，查看自己的任务排名。

Understanding tasks

Evaluation on XGLUE

https://arxiv.org/abs/2004.01401

Generation tasks

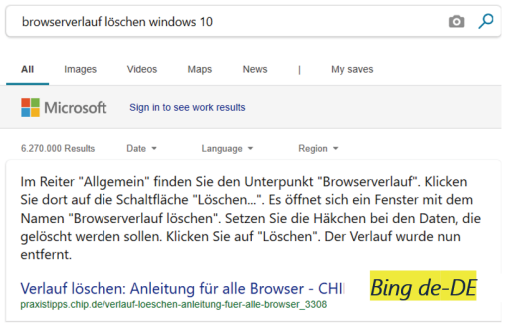
Task	Model	ar	bg	de	el	en	es	fr	hi	it	nl	pl	pt	ru	sw	th	tr	ur	vi	zh	AVG
NER	M-BERT	-	-	69.2	-	90.6	75.4	-	-	-	77.9	-	-	-	-	-	-	-	-	-	78.2
	XLM-R_{base}	-	-	70.4	-	90.9	75.2	-	-	-	79.5	-	-	-	-	-	-	-	-	-	79.0
	$Unicoder_{LC}$	-	-	71.8	-	91.1	74.4	-	-	-	81.6	-	-	-	-	-	-	-	-	-	**79.7**
POS	M-BERT	52.4	85.0	88.7	81.5	95.6	86.8	87.6	58.4	91.3	88.0	81.8	88.3	78.8	-	43.3	69.2	53.8	54.3	58.3	74.7
	XLM-R_{base}	67.3	88.8	92.2	88.2	96.2	89.0	89.9	74.5	92.6	88.5	85.4	89.7	86.9	-	57.9	72.7	62.1	55.2	60.4	**79.8**
	$Unicoder_{LC}$	68.6	88.5	92.0	88.3	96.1	89.1	89.4	69.9	92.5	88.9	83.6	89.8	86.7	-	57.6	75.0	59.8	56.3	60.2	79.6
NC	M-BERT	-	-	82.6	-	92.2	81.6	78.0	-	-	-	-	-	79.0	-	-	-	-	-	-	82.7
	XLM-R_{base}	-	-	84.5	-	91.8	83.2	78.2	-	-	-	-	-	79.4	-	-	-	-	-	-	83.4
	$Unicoder_{LC}$	-	-	84.2	-	91.7	83.5	78.5	-	-	-	-	-	79.7	-	-	-	-	-	-	**83.5**
MLQA	M-BERT	50.9	-	63.8	-	80.5	67.1	-	47.9	-	-	-	-	-	-	-	-	-	59.5	55.4	60.7
	XLM-R_{base}	56.4	-	62.1	-	80.1	67.9	-	60.5	-	-	-	-	-	-	-	-	-	67.1	61.4	65.1
	$Unicoder_{LC}$	57.8	-	62.7	-	80.6	68.6	-	62.7	-	-	-	-	-	-	-	-	-	67.5	62.1	**66.0**
XNLI	M-BERT	64.9	68.9	71.1	66.4	82.1	74.3	73.8	60.0	-	-	-	-	69.0	50.4	55.8	61.6	58.0	69.5	69.3	66.3
	XLM†	73.1	77.4	77.8	76.6	85.0	78.9	78.7	69.6	-	-	-	-	75.3	68.4	73.2	72.5	67.3	76.1	76.5	75.1
	XLM-R_{base}	72.1	77.5	77.0	75.9	84.6	79.2	78.2	69.8	-	-	-	-	75.5	64.7	71.6	72.9	65.1	74.8	73.7	74.2
	$Unicoder_{SC}$	68.5	73.2	71.6	71.6	82.9	75.0	74.7	66.0	-	-	-	-	70.6	64.1	67.0	68.7	62.5	71.2	69.7	70.5
	$Unicoder_{LC}$	73.9	78.5	78.2	77.3	85.4	79.8	79.2	70.1	-	-	-	-	76.7	67.4	71.8	73.8	66.3	75.9	74.7	**75.3**
PAWS-X	M-BERT	-	-	82.9	-	94.0	85.9	86.0	-	-	-	-	-	-	-	-	-	-	-	-	87.2
	XLM-R_{base}	-	-	86.9	-	94.4	88.0	88.7	-	-	-	-	-	-	-	-	-	-	-	-	89.5
	$Unicoder_{LC}$	-	-	87.4	-	94.9	88.8	89.3	-	-	-	-	-	-	-	-	-	-	-	-	**90.1**
QADSM	M-BERT	-	-	60.3	-	68.3	-	64.1	-	-	-	-	-	-	-	-	-	-	-	-	64.2
	XLM-R_{base}	-	-	65.8	-	71.7	-	68.3	-	-	-	-	-	-	-	-	-	-	-	-	**68.6**
	$Unicoder_{LC}$	-	-	64.6	-	71.8	-	68.7	-	-	-	-	-	-	-	-	-	-	-	-	68.4
WPR	M-BERT	-	-	76.6	-	78.1	75.3	74.2	-	70.1	-	-	76.6	-	-	-	-	-	-	64.5	73.5
	XLM-R_{base}	-	-	77.6	-	78.2	76.0	74.4	-	70.7	-	-	77.3	-	-	-	-	-	-	63.9	73.8
	$Unicoder_{LC}$	-	-	77.2	-	78.4	75.7	74.9	-	70.3	-	-	77.4	-	-	-	-	-	-	64.4	**73.9**
QAM	M-BERT	-	-	64.7	-	67.5	-	66.0	-	-	-	-	-	-	-	-	-	-	-	-	66.1
	XLM-R_{base}	-	-	68.1	-	69.3	-	67.8	-	-	-	-	-	-	-	-	-	-	-	-	68.4
	$Unicoder_{LC}$	-	-	68.4	-	69.9	-	68.4	-	-	-	-	-	-	-	-	-	-	-	-	**68.9**
AVG^2_{ZT}	M-BERT																				72.6
	XLM-R_{base}																				75.8
	$Unicoder_{LC}$																				**76.2**
QG	M-BERT	-	-	0.1	-	7.8	0.1	0.1	-	0.2	-	-	0.1	-	-	-	-	-	-	-	1.4
	XLM-R_{base}	-	-	0.1	-	6.0	0.0	0.0	-	0.1	-	-	0.0	-	-	-	-	-	-	-	1.0
	$Unicoder^{sDAE}_{SC}$	-	-	3.0	-	14.0	12.4	4.2	-	15.8	-	-	8.3	-	-	-	-	-	-	-	9.6
	$Unicoder^{sFNP}_{SC}$	-	-	3.7	-	13.9	14.8	4.9	-	17.0	-	-	9.5	-	-	-	-	-	-	-	**10.6**
NTG	M-BERT	-	-	0.7	-	9.0	0.4	0.4	-	-	-	-	-	0.0	-	-	-	-	-	-	2.1
	XLM-R_{base}	-	-	0.6	-	8.1	0.4	0.3	-	-	-	-	-	0.0	-	-	-	-	-	-	1.9
	$Unicoder^{sDAE}_{SC}$	-	-	6.8	-	15.6	9.0	8.7	-	-	-	-	-	7.7	-	-	-	-	-	-	9.6
	$Unicoder^{sFNP}_{SC}$	-	-	7.5	-	15.8	11.9	9.9	-	-	-	-	-	8.4	-	-	-	-	-	-	**10.7**
AVG^2_G	M-BERT																				1.8
	XLM-R_{base}																				1.5
	$Unicoder^{sDAE}_{SC}$																				9.6
	$Unicoder^{sFNP}_{SC}$																				**10.7**

图 12：各个预训练模型在 XGLUE 上的评测结果

上图，我们在 XGLUE 上评测了多个跨语言预训练模型（包括 MBERT、XLM、XLM-R 和 Unicoder），并在表中列出比较结果。可以看到，我们的 Unicoder 在许多理解和生成任务上实现了最佳的平均性能。

English translation: ***where is the largest sugar factory in the world***

English translation: ***The sugar refinery of the Algerian group Cevital produces 2.7 million tonnes of sugar a year, making it the largest refinery in the world. This refinery doubled its exports of white sugar from 377,000 tonnes to 600,000 tonnes in 2012.***



English translation: ***delete browser history windows 10***

English translation: ***In the tab "General" you will find the sub-item "Browser History". Click on the "Delete ..." button there. A window with the name "delete browser history" will open. Check the data you want to delete. Click on "Delete". The history has now been removed.***

图 13：多语言问答技术在 Bing 搜索上的应用

多语言预训练模型可以把英语的模型扩展到其他语言。英语的标注数据比较多，而其他语言往往缺少标注数据。因此，利用多语言预训练模型，可以把英语的模型做好之后，在其它语言上面也能体现出一定程度的问答能力。比如这里展示了问答系统。英文问答数据微调训练的 QA 在法语、德语上也有很好的效果。

en	Input News	if you 're planning a trip to europe , you probably want to check some famous landmarks off your list . but there are certain tourist traps you 're better off missing . susana victoria perez has more .
	Golden Title	do yourself a favor and avoid these tourist traps in europe
	Unicoder$^{xDAE}_{SC}$	tourist traps you should avoid in europe
fr	Input News	alain juppe , candidat a la primaire de la droite , " ne se sent pas engage " par les investitures decidees par le parti les republicains preside par nicolas sarkozy , a affirme jeudi a l' afp son directeur de campagne , gilles boyer . " c' est un processus mene a la hussarde . il n' y a pas de volonte d' equilibre et de rassemblement " , a-t-il denonce , en affirmant que " l' accord politique " entre les differents candidats a la primaire " n' a pas ete respecte " .
	Golden Title	legislatives : juppe " ne se sent pas engage " par les investitures
	Unicoder$^{xDAE}_{SC}$	alain juppe : " ne se sent pas engage " par les investitures
de	Input News	vermutlich zur verteidigung seines reviers hat ein aggressiver bussard in baden-wurttemberg einen radfahrer zu fall gebracht , der sich dabei schwer verletzte . wie die polizei in ludwigsburg am freitag mitteilte , attackierte der greifvogel den 51-jahrigen am vortag auf einem radweg entlang einer landesstraße . der bussard flog demnach so tief auf den radler zu , dass dieser ausweichen musste und sturzte . den angaben zufolge erlitt der mann schwere verletzungen und wurde von rettungskraften in ein krankenhaus gebracht . " aus luftiger hohe , von einem laternenmast aus , beobachtete der raubvogel anschließend die unfallaufnahme " , hieß es im polizeibericht .
	Golden Title	aggressiver bussard bringt radfahrer zu fall
	Unicoder$^{xDAE}_{SC}$	aggressiver bussard in ludwigsburg sturzes radler
es	Input News	despues de la marcha de bruce willis por problemas de agenda , steve carrell le sustituira asi en la nueva pelicula que prepara woody allen . segun informa variety , el actor se une al reparto ya formado por blake lively , parker posey , kristen stewart , jesse eisenberg , jeannie berlin ,corey stoll , anna camp , y ken stott , entre otros . como siempre , los detalles de la trama son aun un secreto aunque el rodaje se encuentre actualmente en marcha . por otro lado , aun no hay fecha de estreno ni distribuidora para la pelicula sin titulo de woody allen . sin embargo , el director tiene aun pendiente de estreno su ultimo filme con emma stone y joaquin phoenix titula da irrational man que se estrenara el proximo 25 de septiembre .
	Golden Title	steve carrell sustituye a bruce willis en la nueva pelicula de woody allen
	Unicoder$^{xDAE}_{SC}$	steve carrell sustituira a steve carrell en woody allen

图 14：多语言新闻标题生成

我们也可以用跨语言或多语言的预训练模型生成新闻标题。同样也是在英语标注集合微调训练之后的系统，也可以生成其他语言的标题。

总结一下。多语言预训练模型缓解了多种语言的资源短缺问题，能够帮助多语言搜索 /QA/ 广告 / 新闻 / 文本摘要 /……低资源神经机器翻译等取得新的提升。

当然，多语言预训练模型也仍然面临许多挑战:

- 首先，最有效的预训练任务仍然是掩码语言模型（在多语种或双语语料库上），我们要拓展出新的任务以便充分利用多语言 / 双语的特点;
- 第二，词汇表比单语言的预训练模型（例如 BERT / RoBERTa）大得多，单语三万，多语就能达到 25 万。这样一来，要学的模型参数就会增加很多，训练的开销更大。而且 25 万词表，表达多语言词汇的覆盖度依然不够;
- 第三、有的语言对有词汇、语法的同源关系，迁移学习效果好，比如英语的微调结果对法语、意大利语、西班牙语比较好，而对汉语的效果不太明显。因此下一步可以考虑在语系内部进行多语言模型训练。

三、预训练模型在多模态任务的应用

下面介绍预训练模型在多模态任务中的应用。

图像 – 语言预训练模型的目的，可以是用于理解或者生成，这里仅介绍理解用的预训练模型。做法如下:

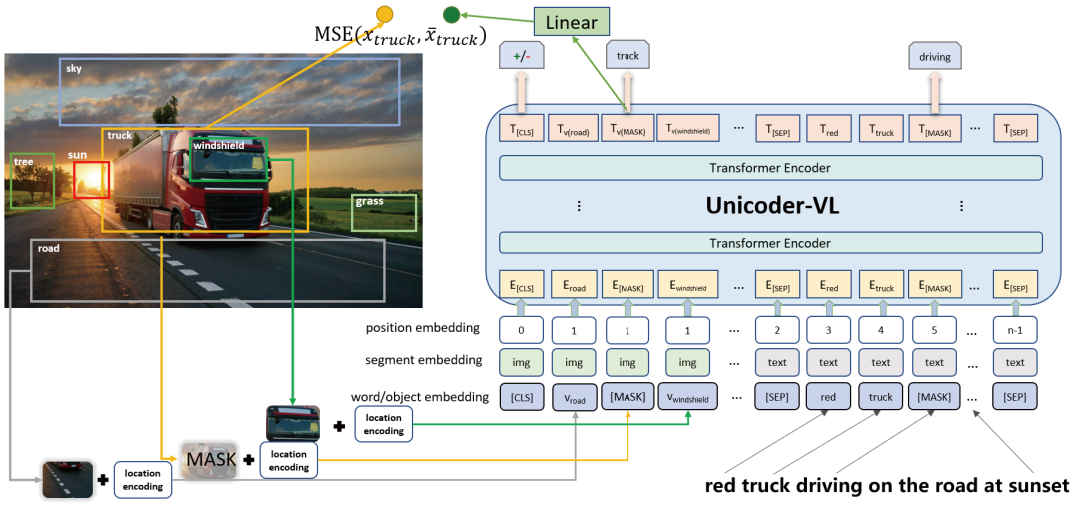


图 15：图像 – 语言预训练模型（ViBERT、Unicoder–VL、VL–BERT、UNITER）

给定一个包含一幅图片和对应的 Caption（就是文字描述）的数据库。比如有这个例子包括图片和对应的文字描述。首先对这个数据库进行预处理，用 Faster–RCNN 得到图片每一个对象 Label 的分布，和对象的输出向量表示（Softmax 之前的输出向量表示）。一个图片的所有对象按照从左到右，从上到下的顺序排列，可以形成一个序列，和文本序列排列在一起。

我们可以用 BERT 方式训练一个预训练模型，比如掩码的方式，盖住文字段的某一个 Token，来预测这个 Token；或者盖住对象序列的某一个对象，来预测这个对象的输出向量与原始向量的相似度。

现有工作基本都基于大致相似的网络结构。我们是最早发表这方面工作的团队之一。我们增加了一个新的训练任务，即预测对象的输出向量，是否可以还原为对象的原始向量，取得了不错的效果。

Model	Text-to-Image Retrieval (Flickr30k)			Image-to-Text Retrieval (Flickr30k)		
	R@1	R@5	R@10	R@1	R@5	R@10
ViLBERT (Lu et al., 2019)	58.2	84.9	91.5	-	-	-
UNITER (Chen et al., 2019)	71.5	91.2	95.2	84.7	97.1	99.0
Unicoder-VL (Li et al., 2020)	73.1	92.3	95.9	88.0	97.3	98.6

Model	Text-to-Image Retrieval (MSCOCO)			Image-to-Text Retrieval (MSCOCO)		
	R@1	R@5	R@10	R@1	R@5	R@10
UNITER (Chen et al., 2019)	48.4	76.7	85.9	63.3	87.0	93.1
Unicoder-VL (Li et al., 2020)	50.5	78.7	87.1	66.4	89.8	94.4

- Pre-training dataset
  - 3,318,333 image-caption pairs from Google's Conceptual Captions

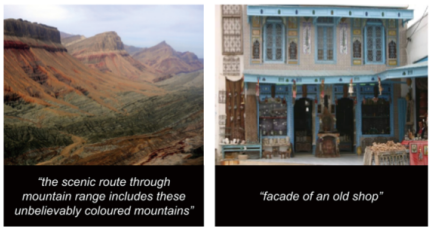


图 16：各个模型的评测结果对比

在 Flickr30K（Image Retrieval 和 Captioning 的数据集），给定 Text 从 1K 的图片（给定）排序得到最优的匹配。反之对 Image2Text 亦然。MSCOCO 是微软提供的数据集，进行 Image2Text 和 Text2Image 两个任务的评测。

预训练数据集大家都一样，大概是 300 多万的 Image–Caption 对。ViLBERT 来自 facebook；UNITER 来自微软产品组；Unicoder–VL 由于增加了新的训练任务（如前述），预训练模型对图片和文本的编码能力有所提升，得到了较好的效果。

Rank	Participant team	Binary	Open	Consistency	Plausibility	Validity	Distribution	Accuracy	Last submission at
1	Human Performance (human)	91.20	87.40	98.40	97.20	98.90	0.00	89.30	1 year ago
2	DREAM+Unicoder-VL (MSRA)	84.46	68.60	91.47	83.75	96.42	3.68	76.04	8 months ago
3	TRRNet (Ensemble)	82.12	66.89	89.00	83.58	96.76	1.29	74.03	2 months ago
4	Kakao Brain	79.68	67.73	77.02	83.70	96.36	2.46	73.33	11 months ago
5	AIOZ (Coarse-to-Fine Reasoning, Sing)	81.16	64.19	90.96	84.81	96.77	2.39	72.14	5 months ago
6	270	77.50	63.82	86.94	83.77	96.65	1.49	70.23	11 months ago
7	NSM ensemble (updated)	80.45	56.16	93.83	84.16	96.53	2.78	67.55	9 months ago
8	TRRNet (Single)	77.91	50.22	89.84	85.15	96.47	5.25	63.20	1 month ago
9	NSM single (updated)	78.94	49.25	93.25	84.28	96.41	3.71	63.17	9 months ago
10	LXMERT (LXR955, Ensemble)	79.79	47.64	93.10	85.21	96.36	6.42	62.71	11 months ago

1/89

图 17：GQA 评测结果

GQA 是来自斯坦福的一个视觉推理和问答评测集：给一个图片和一个问题，从一个固定的答案候选集合（3000 条）中找出一个最优的结果。一般采用的神经网络设计，会将问题和 Image 作为输入，在网络输出层使用 Token[CLS] 通过前向网络和 Softmax，对 3000 个答案候选进行排序。目前我们提交的结果排在第一名。

我们首先使用了预训练模型，发现结果都要好于不用；其次，Unicoder-VL 模型由于前述的原因（比如增加了训练任务），体现了较好的预训练性能；另外，我们对 Query Understanding 进行了 Parser，得到了逻辑表达式，并表征了 Entity 之间的逻辑关系，在网络输入端，我们增加了这个逻辑表达的序列用于微调，从而得到了比较好的效果。

这里有很多有趣的应用，比如给定一个 Query，找到最相关的 Image。

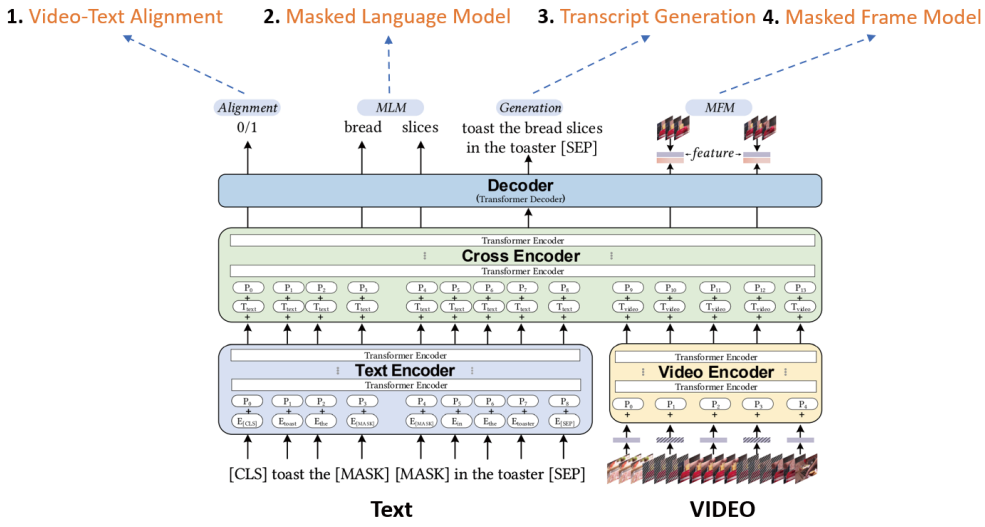


图 18：视频 – 语言预训练

同样的，也可以对 Video-NL（视频 – 自然语言）对进行预训练，对于每一个 Video 片断和对应的 NL 描述（语音识别结果），我们排列在一起，Vdieo 可以按时序进行切分，比如说每一秒切分一下，通过一个 S3D 的工具，使每一个 Video Clip 输出一个向量表示，这样就得到一个 Video 的编码，文本的序列和 Video 的序列排列在一起，进入一个 Transformer，Transformer 可以三层、四层或更多层，后面跟一个解码层，这样利用 Encoder-Decoder 结果来做预训练。这里使用了四个任务进行预训练：

- Video-text alignment 任务用来判断输入的 Video Clip 和 Text 是否相互描述。
- Masked language model 任务用来预测 transcript 里面被 mask 掉的单词。
- Transcript generation 任务基于输入的 Video Clip，生成对应的 Video Transcript，这时候还有 NL 段置空了。
- Masked frame model 任务用来预测被 mask 掉的 Video Clip 对应的 Video Feature Vector。

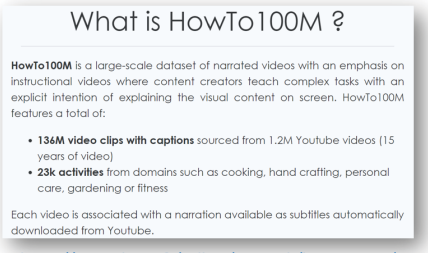
我们把 Unicoder-VL 扩展到 Video，用上述的方法进行训练。跟其他的工作比较，我们把理解和生成放在一个预训练模型里面来做，这样既可以做理解，也可以做生成。

Pre-training Data:

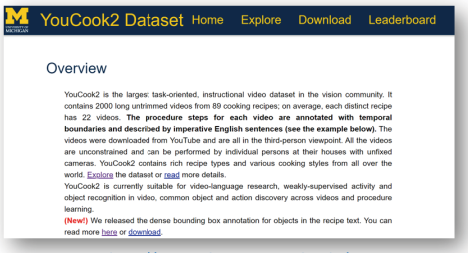
1. HowTo100M: 136M video clips with captions from 1.2M Youtube videos.

(Research) Evaluation Data:

1. YouCook2: 2000 long untrimmed videos from 89 cooking recipes.
2. MSR-VTT: 10K web video clips with 41.2 hours and 200K clip-sentence pairs



https://www.di.ens.fr/willow/research/howto100m/



http://youcook2.eecs.umich.edu/

图 19：视频 – 语言数据集

预训练的语料，目前大家都采用 HowTo 100M，作为预训练的语料，它是从 Youtube 上抓取的 1.2M 视频，切分成 136M 的视频片段，根据时间轴配上文字说明（Youtube 自带的）。下游任务的微调，使用了 YouCook2 的菜谱视频，还有一个微软发布的 Video-Caption 数据集 MSR-VTT。

下游任务包括视频搜索和视频 Caption 生成。

- 检索任务：给定 NL 的 Query，从一个固定视频片段中搜索最匹配的视频片段。
- Caption 任务：给定一段视频，加上 Transcript，生成对应的 Caption。

和 Google 的 VideoBERT、CBT 还有百度的 ActBERT 相比较，我们的模型取得了较好的效果。

它还有很多应用，比如说可以分成两个任务，一个是 Video 的 Segmentation，另一个是对每个 Segmentation 做一个文字的描述。大家看一段比较长视频的时候，可以快速定位到自己感兴趣的部分。

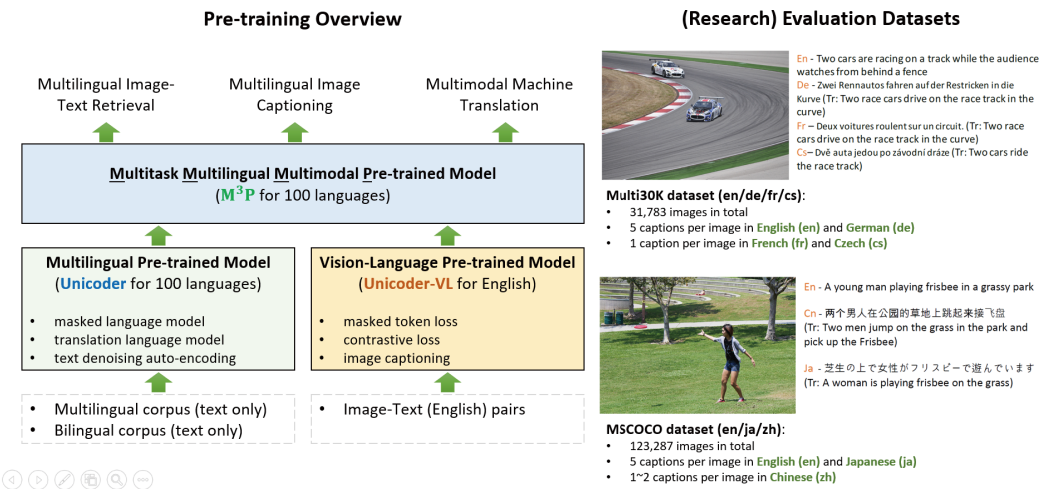


图 20：多语言图像 – 语言预训练

使用同一个 Transformer 模型去融合多语言预训练任务和多模态预训练任务，通过多任务学习和参数共享，可以获得多语言、多模态的能力。这里使用 Multi30K 和 MSCOCO 作为验证测试集。这里说明一下，上图左边是多语言单模态预训练，右边是单语言多模态预训练。它们共享一个 Transformer（十二层），进行多任务的训练，训练任务和前面所述一样。它可以支持多语言、多模态的许多人物，例如多语言 – 图片搜索，但是只需要一种语言的微调数据，就可以支持多语言的搜索； 还有多语言的图片 Caption 生成，给一个图片，生成多语言 Caption，比如同时输出比如说中、英等多种语言的 Caption； 还可以支持多模态的机器翻译，给一个图片，和一种语言的 Caption，我们就可以得到另外一种语言的 Caption。目前 Downstream 数据集中有 Multi30K，一个图片可以配有四种语言的 Caption；MSCOCO 数据集中一个图片有三种语言 Caption。

将上述方法分别在多语言图文搜索（Image2Txt 和 Txt2Image）、多语言图片说明、多模态机器翻译上进行验证。对英文的标注集合进行微调，而对其他语言进行测试。虽然英文上的效果有所下降（这是由于目前模型是针对 100 语言而不仅仅是英语），但对于其他缺乏足够训练语料的语言来说，性能却提升明显。左边的例子是 Image Captioning，这是一个多语言的 Caption 输出，比如输出中文； 右边的例子是多模态机器翻译，给定图片和英文的 Caption，输出多语言（例如法语和德语）的 Caption 翻译。

(en): a view of earth from space
(zh): 从太空看地球的景象

(en): a volcano erupting with lava
(zh): 火山喷发出熔岩

图 21： 图像捕捉

这里有一些应用，比如说我们可以做多语言的 Captioning，给定一个图片，我们想生成哪一种语言的 Caption，就可以生成哪一种语言的 Caption。

四、总结

多模态预训练模型总体来讲尚处于初期阶段。遵循大多数 NLP 预训练模型，将 Transformer 机制，从有限的图像 / 视频 – 语言数据集中学习联合表示，可以计算图像 / 视频片段和文字描述的距离，并实现图像 / 视频 – 文字之间的转换。多模态预训练模型，虽然刚刚开始，还不成熟，但是已经在图像 / 视频的搜索，以及生成文字描述等任务中显示出不错的前景。

当然，多模态预训练模型还仍然面临许多挑战:

- 首先，图像 / 视频 – 语言对的数据的大小仍然比自然语言语料库小得多；

- 第二，CV 仅仅用于特征提取，目前并没有将 CV 模型和 NLP 模型共同训练。当然目前没有好的算法，而且训练的 cost 非常大；
- 第三，就是与之有关的，CV 的图像识别，目前的类别仅限于 1000 类左右，对真实场景的覆盖不够，而且识别的精度也不够。导致预训练的输入信号天然带有误差；
- 第四，对于多模态预训练模型，目前都是用 Transformer 机制，但是它的代价比较大，而且是否最合适对图像 / 视频 – 文字建立关联，还需要进一步探索；
- 第五，图片和视频的预训练模型也不一样，由于视频有时序，因此视频的分割，按照固定时长分割，缺乏逻辑意义。而且视频的 Token 会比 NL 多很多，导致训练的代价比图片和文字的预训练大很多。

NLP 在大数据、大模型、神经网络框架下取得很好的进展。预训练 + 微调形成目前可扩展的解决方案。预训练模型在多语言任务中，在 Rich–Resource 模型的训练结果可以迁移到 Low–Resource 语言任务中，减轻了语言数据不足的问题。预训练模型在多模态任务中是一个有巨大探索空间的全新的领域。图片或视频的预处理，训练任务的设计都有很多有趣的研究。

总体来讲，预训练模型会带来新的研究机会。未来我们需要探索自然语言处理中如何利用常识和知识，来增强推理机制，改善可解释性，而对于预训练模型，我们要考虑新的自学习方法，新的体系结构，以及模型压缩和快速解码等等，这些都是值得进一步探索的课题。

小米集团语音首席科学家 Daniel Povey：可微分的加权有限状态机及其机器学习应用

整理：智源社区 李维

Daniel Povey 本次演讲的主题是《可微分的加权有限状态机及其机器学习应用》。

Daniel Povey，开源语音识别工具 Kaldi 之父，前约翰霍普金斯大学语言与语音处理中心研究型副教授，现任小米集团语音首席科学家。

在演讲中，Daniel Povey 首先指出了当前版本 Kaldi 的一些缺陷，并针对于此提出了一些下一代 Kaldi 发展方向的战略构想；其次，就加权有限状态机这一关键技术以及其在下一代 Kaldi 中如何应用进行了阐述；在报告的尾声，Daniel 历数传统确定化算法的优缺点并条陈了其所提出算法的主要思想。

一、Kaldi 及其下一代

Kaldi，得名于传说中发现了咖啡树的埃塞俄比亚牧羊人，其诞生于 2009 年约翰霍普金斯大学（Johns Hopkins University）的一个名为“新语言和新领域的低开发成本和高质量语音识别”的研讨会。作为语音识别领域的后起之秀，Kaldi 已被工业界和学术界的从业者所广泛接受，俨然成为当前最流行的开源语音识别工具。Kaldi 主要使用 C 及 C++ 进行开发编写，在此之上使用 Bash 和 Perl 以及 Python 脚本调用 C++ 代码进行工具开发。

Kaldi 有着与 HTK 相仿的目标和受众，拥有很多处理实际任务的实例以及大量可以复用的脚本是其广受欢迎的众多原因之一，其鲜明特色主要包括:

1）与有限状态传感器（FSTs）的代码级集成；
2）广泛的线性代数支持，包括一个封装了标准的 BLAS 和 LAPACK 例程的矩阵库；
3）可扩展设计；
4）开放式许可。

Kaldi 的优点不可否认，但也有十分复杂以及没有专长技能作为前提则不易学会的缺点。此外，因为 Kaldi 本身不支持模型量化，故很难在手机上实现产品化。虽说 Kaldi 使用的是自己的深度学习框架，但这个框架并不容易使用。Daniel Povey 也在本次报告中直言不讳地指出“尽管 Kaldi 拥有自己的神经网络框架，但其通用程度却不及 PyTorch 和 TensorFlow”，故他便有了将 PyTorch 应用到下一代 Kaldi 深度神经网络中且允许在 PyTorch 和 TensorFlow 之间实现灵活切换的想法。如若这个想法在下一代 Kaldi 中得以实现，那将使得 Kaldi 与标准框架 PyTorch 和 TensorFlow 实现更好的结合。

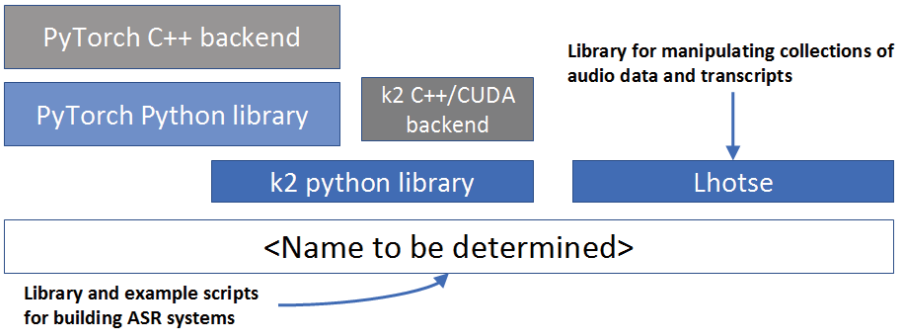


图 1：下一代 Kaldi 框架图

Daniel Povey 表示，下一代 Kaldi 将非常不同，几乎没有与现有 Kaldi 通用的代码，他希望下一代 Kaldi 能实现以下目标:

1) 用少量代码就实现像连接性时许分类算法 (CTC) 这样的功能;
2) 轻松有效地整合“离散”信息源，如词汇和音素序列信息等;
3) 将“传统”自动语音识别 (ASR) 解码与 PyTorch 模型以简单的方式集成;
4) 有效地操作序列和序列集合;
5) 使用通用而不是过于具体的工具来执行操作。

尽管下一代 Kaldi 注定会有所变革，但 Daniel 表示在创造一系列工具用以实现这一目标之前，有限状态机是一个不得不解释的概念。

二、有限状态机

2.1 何为有限状态机

有限状态机，也被简称为 FSA (Finite State Acceptors)，其主要被用以研究具有有限内存的计算过程，并根据一定的规则响应外界输入值，对研究对象的状态变化进行枚举，得出状态变化序列。作为一种依据对象行为建模的工具，其被广泛应用于电路设计、软件工程、网络协议和语言研究等计算机科学中的众多领域。如图 2 所示即为一个极为简单的有限状态机，图中两个圆圈称为节点，用以表示两种状态，并分别用 0 和 1 记之。在任意时刻有限状态机均处于有限状态集合的某一状态，其中有限状态集合可被一般地表示为 { 状态 1，状态 2，…，状态 m}，m 需为一有限数，这是其之所以被称之为有限状态机的原因和要求。

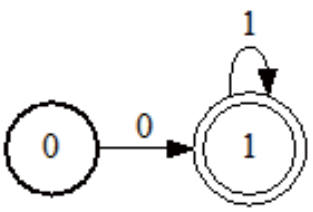


图 2：一个简单的有限状态机

从图中不难看出，从每个节点或是状态出发都有一条边，复杂的有限状态机则有多条边并通向多种状态。处于某一状态的有限状态机在获得输入字符时会引起状态转换，将从当前状态转换到下一状态或是仍然保持当前状态不变化，其依据就是输入字符是否跟该节点出发的某条边的内容一致。例如，若有限状态机目前正处于状态 0 且输入字符也为 0，那么该有限状态机则会从状态 0 进入状态 1。对于状态 1 而言，若输入的字符为 1，则其会保持当前状态不变。倘若当前状态不存在与输入字符对应的输出边，那么该有限状态机就会进入消亡状态 (Doom state)。例如，当有限状态机处于状态 0 时输入字符 1 或是在状态 1 时输入字符 0。此外，对图示有限状态机而言，输入任何非 0 和 1 的字符均会导致该状态机进入消亡状态。为方便和直观，一般绘制如表 1 所示状态转换表。

表 1：有限状态机转换表

输入字符	0	1
状态 0	1	消亡状态
状态 1	消亡状态	1

Daniel Povey 指出，有限状态机中有两个特殊的状态，被称为起始状态和结束状态。当有限状态机开始工作，输入字符会导致状态机的状态不断变化，但只要最后输入的那个字符使得状态机能转化到结束状态，那么该状态机就会结束工作，识别出所输入的所有字符序列。例如，在图 1 中假定状态 0 和状态 1 分别为起始和结束状态，那么该有限状态机就会接收“0”，“0 1”，“0 1 1”，“0 1 1 1”等字符串。由于该状态机设置得较为简单，故其在该种情况下只能接受类似的有限字符串。倘若输入的字符串为“0 1 0”，那么由于字符 0 导致状态机会进入消亡状态，故该字符串将被状态机所拒绝。

除了有限状态机之外，加权有限状态机作为有限状态机的一种特殊形式亦是构建快速语音识别系统的主流技术。

2.2 加权有限状态机

在语音识别领域有着广泛应用的加权有限状态机 (Weighted Finite State Acceptors，WFSA) 事实上是以有限状态机为蓝本，顾名思义，在其输出边或弧上拥有权值信息。

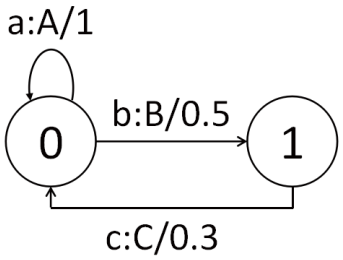


图 3：一个简单的加权有限状态机

如图 3 所示即为一个简单的加权有限状态机，不难发现其是一个有向图。连接各状态之间的状态转移弧上分别标示着输入符号、输出符号以及相应的权值信息。图中所示的输入和输出符号有所不同，但在现实情况中允许一个加权有限状态机具有相同的输入和输出符号。图中的状态机会在输入小写英文字母之后输出相应的大写字母，而在实际的语音识别应用中，可能是以发声的声韵母作为输入符号，并以汉字或是词语作为输出。

一般而言，加权有限状态机除了以状态转移弧和结束状态赋有权值为显著特点之外，还需要半环代数理论作为支撑。一个简单的半环代数结构通常由元素集合、两个二元运算和两个基本单元构成，可被形式化表示为 $(K,\oplus,\otimes,0,1)$。应当特别指出，这里的 0 和 1 并不是真正的实数 0 和 1，而是代指零元和幺元。半环代数需满足的公理和条件有加法的结合律和交换律、乘法的结合律、分配律等。具体如表 2 所示:

表 2：半环代数所满足的公理

公理	形式
加法结合律	$(x\oplus y)\oplus z=x\oplus(y\oplus z)$
加法交换律	$x\oplus y=y\oplus x$
乘法结合律	$(x\otimes y)\otimes z=x\otimes(y\otimes z)$
分配律	$(x\oplus y)\otimes z=(x\otimes z)\oplus(y\otimes z)$ $x\otimes(y\oplus z)=(x\otimes y)\oplus(x\otimes z)$
幺元	$x\oplus 0=0\oplus x=x$ $x\oplus 1=1\oplus x=x$
零元	$x\otimes 0=0\otimes x=0$

表 3 则显示了一些常用的半环。在语音识别领域使用频率较高的半环有 Log 半环（Log semiring）和热带半环（Tropical semiring）。在 Log 半环中，权值被当作负对数概率来处理，概率在并行路径上求和。在热带半环中，权重值被视为类似于成本之类东西（例如，距离），并以使成本最小化作为合并并行路径时的语义。

表 3：一些常用的半环

半环	集合	$\oplus$	$\otimes$	0	1
布尔半环	$\{0,1\}$	$\vee$	$\wedge$	0	1
概率半环	$R_+\cup\{+\infty\}$	$+$	$\times$	0	1
Log 半环	$R\cup\{-\infty,+\infty\}$	$-\log(e^{-x}+e^{-y})$	$+$	$+\infty$	0
热带半环	$R_+\cup\{+\infty\}$	$\min$	$+$	$+\infty$	0

在谈及加权有限状态机在下一代 Kaldi 中的应用时，Daniel Povey 表达了以下想法:

1）在下一代中，会将权重与有限状态机的结构分开，并尽可能忽略权重；
2）用相反的符号存储权重，比如负定的 Cost 或 Logprob 之类的，并称之为“分数”；
3）获取权重信息的操作支持两种类型：一个相当于“热带半环”（取最大值），另一个相当于“Log 半环”（取 Log Sum Exp 或 Soft Max）；
4）或许将采用更一般的权重类型，但其或只能通过标量与核心算法交互，例如 Pruning 算法。

三、有限状态机确定化

有限状态机确定化（FSA determination）是对有限状态机的基本操作之一，其他基本操作还有合并操作、组合操作以及权重推移操作等等。对一个有限状态机进行确定化操作的目的是为去除原始有限状态机的冗余，得到一个等效的确定的有限状态机，使得该状态机能接收与原始状态机一样的路径集。

3.1 确定化操作的意义

对于确定化有限状态机的每一个状态来说，同一个输入符号有且只有一个转移弧。例如，在图 4 的原始加权有限状态机中，对于状态 3 来说有两条具有同样输入和输出字符但不同权重的状态转移弧与之对应，而被确定化之后得到的图 5 所示的加权有限状态机则不会出现类似情况。那么由此可见，进行确定化操作之后的加权有限状态机相较原始状态机而言就具有了非冗余性。当给一个确定化的有限状态机输入符号序列时，该状态机最多只有一条路径与输入字符序列相对应，如此以来搜索算法的时间和空间复杂度就会被降低，这也是确定化操作的作用之所在。

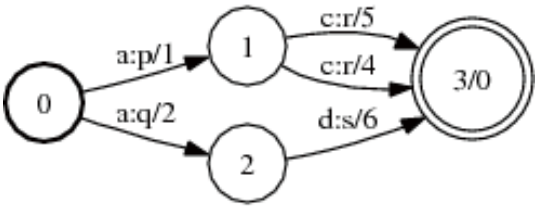


图 4：原始加权有限状态机

（http://www.openfst.org/twiki/bin/view/FST/DeterminizeDoc）

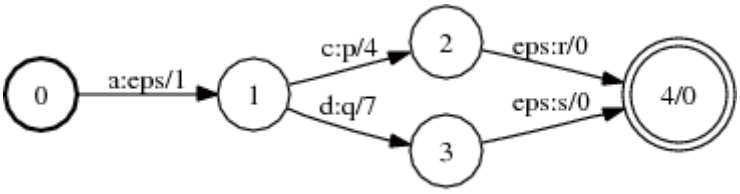


图 5：进行确定化操作之后的加权有限状态机

（http://www.openfst.org/twiki/bin/view/FST/DeterminizeDoc）

Daniel Povey 在谈及有限状态机确定化时指出其是一个非平凡的且具有不规则结构的算法，并希望就该算法而言能做到以下几点：

1）期望能够在 GPU 上将其实现；
2）可阐述计划在下一代 Kaldi 中使用的一些数据类型和抽象概念；
3）期望下一代 Kaldi 比有限状态机更加通用；
4）试图扩展能在 GPU 上实现轻松编码的范围。

3.2 确定化算法间的比较

Daniel Povey 在介绍他的基于热带半环的确定化算法前列举了一些传统算法的特点和不足，其中包括传统的不加权算法以及加权算法。关于传统的不加权算法，Daniel Povey 认为它有如下特点：

1）输出中的每个状态对应于输入中的一个状态子集；
2）输出中的初始状态对应于 {0}，即输入中的起始状态；

3）需维护要处理的输出状态队列；
4）需维护从输入状态 ID 到输出状态 ID 的映射。

关于传统的加权算法，Daniel Povey 指出：

1）加权后映射是从输入状态 ID 的加权集合到输出状态 ID；
2）权重需要以某种方式标准化，例如，在传统的热带半环中，权重要使最小成本为 0；
3）标准化删除的额外权重（Extra weight）将成为输出弧上的权重；
4）会破坏并行性并具有糟糕的浮点数舍入属性。

这里所谓的并行化包含（1）批量确定状态；（2）首批仅有一个元素；（3）批量处理的规模取决于有限状态机拓扑结构；（4）可在实际中的某个小批量中并行处理多个有限状态机等特性。

关于 Daniel Povey 所提出的确定化算法，他总结到：

1）将状态子集表示为（起始状态，符号序列）这种形式是映射的关键；
2）子集是从起始状态通过符号序列可到达的状态的集合；
3）标准化过程包括删除符号序列前缀和推进初始状态；
4）输出弧上的“分数”将是标准化中删除的输入弧的“分数”之和；
5）需要进行确定化操作的有限状态机将更少。

3.3 数据结构：列表套列表

在报告中 Daniel Povey 还提及一种名为列表套列表（ListOfList）的数据结构，其在存储一组固定类型的（大小可变）列表时，采用类似图 6（a）所示将其连接在一起的形式，并按图 6（b）的形式存储与每个子数组开头相对应的索引，再加上一个子数组结束后的索引。此外，利用此思想还可以处理三维及多维不规则数组，只需额外增加索引数量即可，图 6（c）所示。通常来讲，列表套列表型结构有两种检索方式：其一为分层索引（Hierarchical indexing），比如 list[i][j]，就像通常索引向量 <vector<X>> 一样；其二为平面索引（Flat indexing），比如 list.elems[k]，其中 list.elems 是元素的展开列表。

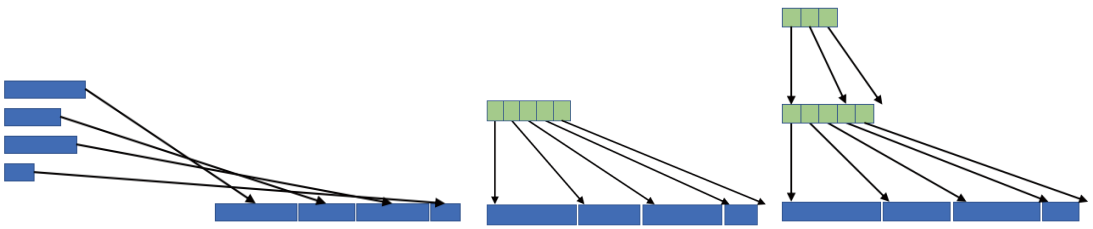


图 6：列表套列表

四、总结

在报告的尾声，Daniel 表示其已经以有限状态机确定化操作为例，设计用于处理不规则结构的并行计算框架，

愿景是使其能够拓展 PyTorch 乃至 TensorFlow 等框架平台。此外，他还提到已经开始了关于非 GPU 实现的工作，并希冀该工作不仅局限于设计一个自动语音识别（ASR）工具包。

从有限状态机到加权有限状态机，从 Kaldi 到其下一代，技术的不断发展进步促使着工具的不断更新演变，而工具使用中所暴露出的问题又是下一次技术进步的动力和源头。正是在这样的互相促进中，Daniel Povey 博士及其团队酝酿着 Kaldi 这一语音识别工具持续向更好、更优、更加完美的方向发展。但“金无足赤，人无完人”，正如 Kaldi 作为“后浪”曾席卷了它的前任们一样，未来会不会有另一工具成为 Kaldi 的下一个继任者呢？会不会有另一巨匠能够站在 Daniel Povey 的肩膀之上呢？让我们拭目以待！

圆桌论坛 AI 新疆域：多模态自然语言处理

整理：智源社区 亓麟

语音、文本、图像等单一模态领域，在以深度学习为主的机器学习算法的推动下，已经取得了巨大的成功。然而在复杂情况下，完整的信息会同时涉及多种模态；利用单一模态信息来完成任务，往往力不从心。因此，近年来多模态机器学习研究逐渐发展起来，并取得了许多重大进展，成为了人工智能的一个重要分支。但多模态研究仍处于起步阶段，其中既面临着巨大的挑战，也存在着巨大的机遇。

那么，在自然语言处理领域，多模态研究又将怎样发展呢？围绕这一问题，6 月 22 日，在第二届智源大会上举行的“语音与自然语言处理专题论坛”中，由京东集团技术副总裁、智源学者何晓冬主持召开了“**AI 新疆域：多模态自然语言处理**”的圆桌论坛，斯坦福人工智能实验室（SAIL）主任 Christopher Manning、华盛顿大学电子与计算机工程教授 Mari Ostendorf、微软亚洲研究院副院长周明、小米集团语音首席科学家 Daniel Povey 等在线上汇聚一堂，就多模态自然语言处理发展中的关键问题进行了深度对话。

一、构建多模态知识库很重要

何晓冬：随着研究者们把目光聚焦在纯文本之外的其它模态的信息，自然语言处理领域迎来的新的机遇和挑战，人们很希望能从多模态数据中获益。另一方面，在过去的几年当中，人们越来越关注对数据的研究，并开始在大规模数据集上预训练。规模庞大的数据虽至关重要，但在多模态多轮对话等复杂的应用场景下，光靠大量的文本数据是不够的，还需要尽可能多的所谓的“知识”。那么“**多模态知识驱动的自然语言处理**”这一关键问题，接下来的几年里会有怎样的技术突破和发展呢？最近的突破是大规模预训练模型 BERT，以及其它大量数据注入的模型和处理大规模数据的新算法。那么多模态**知识驱动的自然语言处理**是否会带来类似的突破呢？

Christopher Manning：在 60、70、80 年代研究者眼里，一个很自然的想法就是如何用具有知识的算法来得到更好的智能推断的效果。但在当时建立一个完备的基于知识库的系统是很困难的。尽管如此，还是有人不断的在建立完备的知识库上不断努力。现在看来，很多人相信这样的想法似乎是错误的，因为目前我们可以在一个领域内通过大量的训练数据得到不错的知识表示效果。然而，最近许多多模态相关的研究证明，**超越文本的多模态知识库是非常重要且困难的**。我们想要的知识并不是像从百科全书中抽取词条那么简单，例如要判断一个人是否喜欢牛仔裤，需要了解关于这个人本身的许多背景知识，这些知识可以从对话中提取，也可以从其他模态的数据中获取。**如果能很好的获取感兴趣内容的多模态的完整知识，那么将对多轮对话领域发展起到重要作用**。

何晓冬：谢谢 Christopher 教授精彩的分析，这让我想起 Mari 在演讲中讲到：**自然语言处理中常用的“背景信息”应该是随着时间和状态发生变化的**，而非一个静态的知识表示，Mari 关于语言背景信息的定义和你说的用户相关信息很相似。Mari 如何看待这一观点呢?

Mari Ostendorf：我同意 Christopher 教授的观点，**用户相关的背景信息用于建模是很重要的**，人们日常在谈论某一件事情的时候往往综合了许多不同的信息。在需要快速反应的对话系统中，往往需要从一个对话场景快速切换到另一个场景，好的知识表示有助于快速得到信息。想要把任何东西都用一大串文本来表示是不现实的，用科学的知识表示显得尤为重要。**好的知识表示应当具有“进化”能力**，能够随着时间变化。当然了，知识表示存在一定的信息冗余，人们可以有选择地运用这些知识表示。

二、多模态数据如何驱动 NLP 的发展

何晓冬：Mari 教授提到**知识并不一定是必须有用的，但却是我们必须具备的，可以有选择性的使用**，这个观点非常有趣。与多知识相关的研究也包括了多任务、多语言和多模态学习，这些在不同任务上分布的数据来源非常广泛，但往往结构性不强。这类多模态数据将如何驱动 NLP 领域的发展呢?

周明：知识表示是非常重要的，但同样重要的一点是哪类知识是我们真正需要的。知识可以分为共性的、任务相关的、开放领域等多种类型。我们的语言学知识更依赖于具体的任务。尽管预训练模型可以学习到许多共性的知识，但真正在下游任务上使用的话，还需要进一步用任务相关的数据来训练模型。举个问答系统的例子来说，仅仅靠以往发布的训练数据就可以训练一个不错的模型吗？我想不是的，好的问答系统应当对对话场景有一个比较好的适应，用户满意的不是共性答案，而是那些最适合具体问题场景的答案。**总而言之，从包括视觉、语言等多模态数据中尽可能广泛的获取知识是非常重要的，但更为重要的是如何在特定场景下有选择性的使用这些知识。多模态**预训练就是一个很好的获取跨模态的知识的方式，未来还有很多多模态预训练相关的工作可以做。

何晓冬：周明老师的观点很有启发性，为了抽取出真正需要的知识，把预训练得到的知识和任务相关的知识进行结合更能够适应现实任务的需要。人类的语言内容要通过语音发出，Daniel 是语音方面的专家，您怎么看待多模态知识这个问题呢?

Daniel Povey：在我看来语音信号本身和知识关系不大，因为语音信号的发出是物理过程，知识是无法通过语音信号和语音模型区分的。所以从单纯的语音到知识过程，似乎研究意义不大，但通过语言这一桥梁就可以连

接语音和知识了，**所以语音这一模态的信息更依赖于通过语言来体现**。

三、值得期待的技术突破

何晓冬：人类说出话语的过程实际上是语言表达的过程，也是知识传递的过程。由于知识结构的复杂性，不同的研究方向会有不同的解读。不过从当下的研究进展来看，预训练的确是目前最好的从文本语言中获取知识的手段。超越文本的知识需要新的解决方案，刚才 Mari 提到背景知识用于建模的方法，及知识表示应具有进化能力的观点非常精彩。周明博士则从如何获取有用知识的角度进行了分析。事实上，NLP 领域最近也逐渐从纯文本的研究迈向了多模态研究，例如融合文本和视觉信息。同样随之而来也有许多有趣的应用，例如图片问答、多模态对话系统等等。自然语言处理领域的发展非常十分迅速，不仅带动了许多任务相关领域的进步，也推动了语言模型本身如 BERT 的发展。多模态作为自然语言处理的新的突破口，Manning 博士，在您看来最值得期待的进展和技术突破是什么？

Christopher Manning：多模态确实是一个值得探索的方向，也能看出来有许多有趣的工作值得去做，比如图片标题生成、视觉问答等。我比较期待的发展方向是**从多模态角度出发，综合多种信息来回答一系列问题的智能体的出现**，并能实现多种信息之间的交互，这些信息中相当一部分来自非语言学知识。

何晓冬：事实上，人们已经开始研究 Manning 教授所说的多模态信息交互了 。智源发布的多模态对话数据集和挑战赛正是为了推动多模态信息交互而开展的。刚才 Mari 教授也提到，不同的信号处理能够得到不同的模态数据，不仅仅可以从图像、文本角度出发，也可以从音频本身的频率信息出发获取有用的音频模态信息，Mari 教授可以详细说一些这个思路吗？

Mari Ostendorf：我认为多模态信息除了图像和文本，音频中也存在大量信息，比如音频的韵律对分析一个人说话的情感就非常重要。另外，**多人对话的研究将是一个新的研究方向**。在多人讨论的场景下，准确地识别当前在和哪个人对话是一项必要工作。此外，如何利用更多模态的信息，来更好的实现人机交互也是需要不断努力的方向。**另一个可研究方向是刚才 Manning 教授提到的类人智能体，与智能体交互的时候，智能体应该能和人一样，对周围的环境有一个比较强的视觉辨识能力，也应该对对话内容有一个全面的认识，几种模态之间信息的对齐和筛选是至关重要的**。

何晓冬：在一个非常复杂的场景当中，如果想要实现 Mari 教授所说的，复杂环境下的交互的智能体，那必然就需要许多传感器来获取多种信息，并这些信息进行进一步的区分和汇总。谈到多种信息，我想起周明老师在演讲中提到了多语言学习的相关研究，那假如我们想要一个智能体能够懂得一百种语言，自然就需要跨语言学习，关于多模态信息的跨语言学习研究，我们可以有什么期待呢？

周明：刚才 Mari 教授和 Manning 教授所说的观点我是很赞同的，我从实际产业视角下来看也能得出类似的结论。不过从**产业上的大数据量、深层次模型和大规模应用的要求之下，如何灵活有效的训练多语言和多模态模型是一个至关重要的问题**。数据是模型的第一个关键点，**首先要构建一个具有统一范式的多模态数据库，并不断在有趣的任务上进行尝试**。如何获得足够大量、准确、多方面的多模态数据本身就是一个不小的挑战。**其次，要找到新方法来高效训练具有强适应能力的深度模型**。产业界也很关注用户体验，好的客户服务需要了解客户多方面的信息，好的多模态语言处理也应当利用与语言信息有关的其它信息。当构建了大规模多模态数据集之后，如何对信息进行有效整合，是对研究人员提出的新的挑战。

何晓冬： 从周明博士的分析看来，尽管大家面对的是同样的科学问题，产业界和学术界确实也还有着不一样的要求。那么我想问一下 Daniel 作为产业界的语音处理专家，在处理语音的时候，会不会考虑情感等信息呢?

Daniel Povey： 我对语音识别领域有着挺长时间的研究，开发和维护了语音识别开源工具 Kaldi，目前我们已经能够成功的进行语音到文本的转换。但**音频信息的利用还有很大的前景**。如果能有效的对音频中的音调、音色、韵律等信息进行分析提取，获取到的也将会是很有用的多模态信息。当前对大规模多模态数据的标注面临一些挑战。例如如何对大规模的音频和视频数据进行标注，粒度应当如何，什么样的标签信息是真正有意义的，这些问题都值得去深入探索。

何晓冬： 非常感谢几位专家学者从自己的研究兴趣出发，对多模态自然语言处理的研究做了鞭辟入里的分析。多模态方向的研究从数据构建、建模方法、评估标准、训练算法等多个角度来讲都是一个较新的领域，也是很有发展前景的方向，未来多模态自然语言处理的研究方向将大有可为。