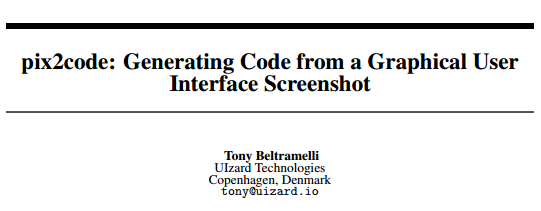
机器之心对这篇论文内容进行了概述性介绍，内容如下：



****摘要：****计算机开发人员经常将设计师设计的图形用户界面（GUI）截图通过编译计算机代码应用到软件、网站和移动应用程序中。在本文中，我们展示了给定图形用户界面图像作为输入，深度学习技术可以被用来自动生成代码。我们的模型能够从单一输入图像中生成针对三种不同平台（即 iOS、Android 和基于 Web 的技术）的代码，其准确率超过 77％。

****引言****

在客户端软件实现基于由设计师设计的图形用户界面（GUI）的过程是开发人员的责任。然而，编写实现 GUI 的代码是耗时的，并且占用了开发人员大量用于实现软件实际特征和逻辑的时间。此外，不同的特定平台用于实现这种 GUI 的计算机语言也不尽相同；从而导致在开发针对多个平台的软件时繁琐而重复的工作（尽管都运用本机技术）。在本文中，我们描述了一个给定图形用户界面截图作为输入，可以自动生成特定平台代码的系统。我们推断，此方法的扩展版本可能会终止手动编程 GUI 的需要。

本文的第一个贡献是 pix2code，一个基于卷积和循环神经网络的新方法，它能够由单个 GUI 屏幕截图生成计算机代码。

本文的第二个贡献是发布来自三个不同平台的 GUI 屏幕截图和相关源代码组成的合成数据集。在本文发表后，此数据集将开源免费使用，以促进今后的研究。

****Related Work（略）****

****pix2code****

给出 GUI 屏幕截图生成代码的任务可以类比于给出场景照片生成文本描述的任务。因此，我们可以将问题分为三个子问题。首先，是一个计算机视觉问题：理解给定场景（即这种情况下为 GUI 截图）并推断图中的对象、身份、位置和姿势（即按钮、标签、元素容器）。第二，是一个语言模型问题：理解文本（即这种情况下为计算机代码）并产生语法和语义正确的样本。最后，通过利用前两个子问题的解决方案生成代码，即运用从场景理解推断出的潜在变量（latent variable）来生成相应文本描述（这里是计算机代码而不是文本）。

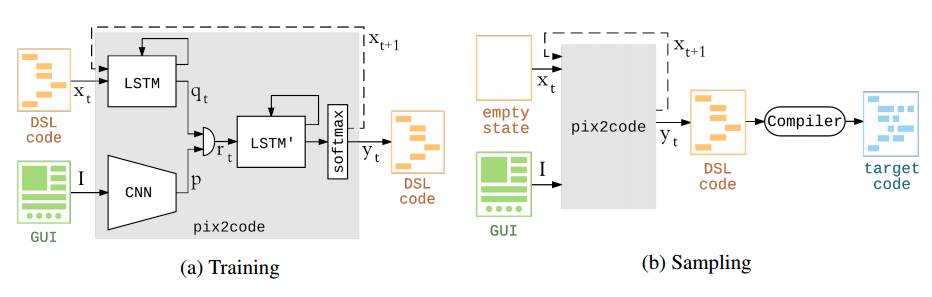


图 1：pix2code 模型的架构概述。训练中，GUI 屏幕截图由基于 CNN 的计算机视觉模型编码；对应于 DSL 代码（领域特定语言）的独热编码（one-hot encode）符号的序列由含有 LSTM 层堆栈的语言模型编码。然后将两个得到的编码向量进行级联并馈送到用作解码器的第二个 LSTM 层堆栈中。最后，用 softmax 层对符号进行单个抽样；softmax 层的输出大小对应于 DSL 的词汇量大小。给定图像和符号序列，模型（即灰色框中的内容）是可微分的，因此在预测序列中的下一个符号时可以通过梯度下降来端到端地优化。在每次预测时，输入状态（即符号序列）都会被更新以包含上次预测的符号。在抽样时，使用传统的编译器设计（compiler design）技术将生成的 DSL 代码编译为所需的目标语言。

****3.1 视觉模型****

CNN 目前是解决各种视觉问题的首选方法，因为它们自身的拓扑结构便于学习训练的图像中丰富的潜在表征 [14,10]。我们通过将输入图像映射到一个学习到的固定长度向量，运用 CNN 来进行无监督特征学习；从而起到如图 1 所示的编码器的作用。

初始化时，输入图像的大小重新定义为 256×256 像素（不保留宽高比），像素值在馈送到 CNN 之前被标准化。不进行进一步的预处理。为了将每个输入图像编码为固定大小的输出向量，我们专门使用卷积步长（stride）为 1 的小 3×3 感受野（receptive field），此方法与 Simonyan 和 Zisserman 用于 VGGNet 的方法相似 [15]。这些操作进行两次，然后运用最大池化进行下采样（down sample）。第一个卷积层的宽度为 32，其后是宽度为 64 的层，最后为宽度 128 的层。最后，使用 2 个大小为 1024 的全连接层运用修正线性单元激活函数（rectified linear unit activation）来完成视觉模型的建模。

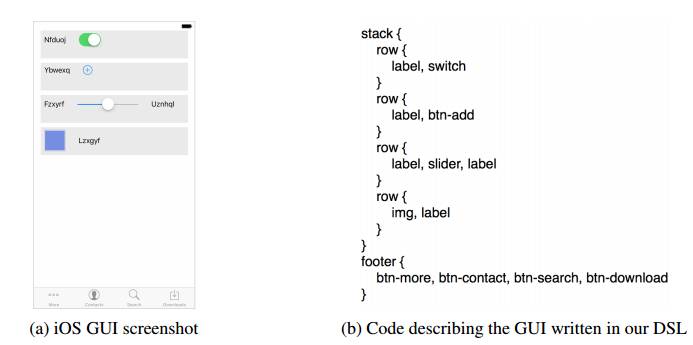
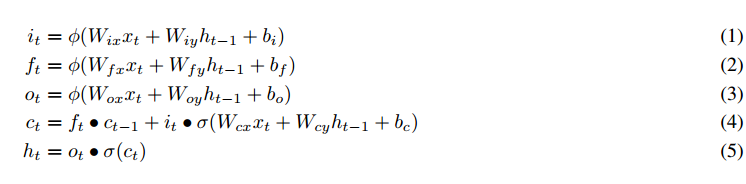


图 2：在 DSL 中编写本机的 iOS GUI 的例子。

****3.2 语言模型****

我们设计了一个简单的 DSL 来描述图 2 所述的 GUI。在该项工作中，我们只对 GUI 的布局感兴趣，只对那些不同图形控件及其相互间的关系感兴趣，因此我们的 DSL 实际上忽略了标签控件的文本值。另外为了缩小搜索空间的大小，简化 DSL 减少了词汇量的大小（即由 DSL 支持的符号总数量）。因此我们的语言模型可以通过使用独热编码（one-hot encoded）向量的离散型输入执行字符级的语言建模，这也就降低了对如 word2vec [12] 那样词嵌入技术的需求，因此也就大大降低了计算量。

在大多数编程语言和标记语言中，元素通常声明为开放符号。但如果子元素或指令包含在一个块（block）内，解释器或者编译器通常需要一个封闭符号（closing token）。如果包含在父元素中的子元素数量是一个变量，那么重要的是构建长期相关性（long-term dependencies）以封闭一个开放性的块（block）。而传统的循环神经网络（RNN）在拟合数据时会出现梯度消失或梯度爆炸的情况，因此我们选择了能很好解决这个问题的长短期记忆（LSTM）。不同的 LSTM 门控输出可由以下方程组计算：



其中 W 为权重矩阵，xt 为在时间 t 的新输入向量，ht−1 是先前生成的输出向量，ct−1 是先前生成的单元状态输出，b 是偏置项，而 φ 和 σ 分别是 S 型激活函数（sigmoid）和双曲正切激活函数（hyperbolic tangent）。

****3.3 复合模型****

我们的模型采用的是监督学习，它通过投送图像 I 和符号 T 的序列 X（xt, t ∈ {0 . . . T − 1}）作为输入，而将符号 xT 作为目标标注。正如图 1 所示，基于 CNN 的视觉模型将输入图像 I 编码为向量表征 p，而基于 LSTM 的语言模型通过将输入符号 xt 编码为中间表征 qt 而允许模型更关注于特定的符号和少关注于其他符号 [7]。

第一个语言模型由两个 LSTM 层、每层带有 128 个单元的堆栈而实现。视觉编码向量 p 和语言编码向量 qt 可以级联为单一向量 rt，该级联向量 rt 随后可以投送到基于 LSTM 的模型来解码同时由视觉模型和语言模型学到的表征。因此解码器就学到了输入 GUI 图像中的对象和 DSL 代码中的符号间的关系，因此也就可以对这一关系进行建模。我们的解码器由两个 LSTM 层、每层带有 512 个单元的堆栈而实现。整个架构可以数学上表示为：

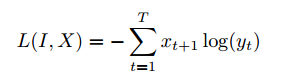


该架构允许整个 pix2code 模型通过梯度下降实现端到端的优化，这样以便系统在看到图像和序列中前面的符号而预测下一个符号。

****3.4 训练****

用于训练的序列长度 T 对长期相关性（long-term dependencies）建模十分重要。在经验性试验后，用于训练的 DSL 代码输入文件由大小为 48 的滑动窗口（sliding window）分割，即我们用 48 步展开循环神经网络。

训练由损失函数对神经网络的权重求偏导数而执行反向传播算法，因此可以最小化多级对数损失而进行训练：



其中 xt+1 为预期符号（expected token），yt 为预测符号。

****3.5 抽样****

为了生成 DSL 代码，我们投送 GUI 图像和 T = 48 符号的序列 X，其中符号 xt . . . xT −1 设置为空符号，而序列 xT 设置为特别的<START>符号。

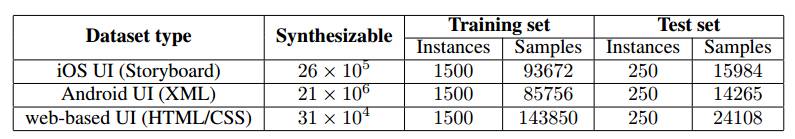


表 1：数据集统计

****试验****

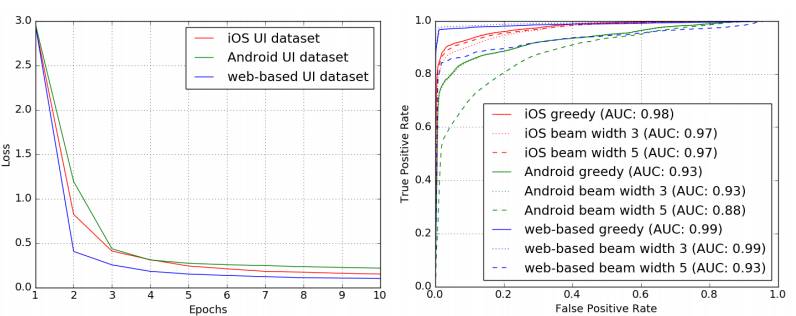


图 3：在不同训练集的训练损失和模型在训练 10 个 epochs 中的抽样 ROC 曲线。

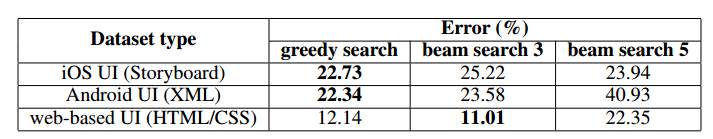


表 2：在测试集（表 1 所述）上的试验结果报告。

图 4、5、6 展示了输入 GUI 图像（样本真值），和由已训练 pix2code 模型生成输出的 GUI。



图 4：iOS GUI 数据集中的试验样本。



图 5：安卓 GUI 数据集中的试验样本。

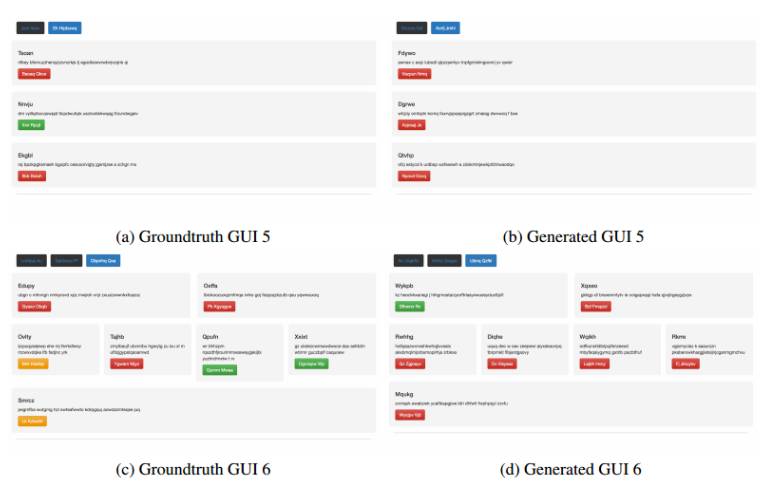


图 6：网页 GUI 数据集的试验样本。

****结语****

在本论文中，我们提出了 pix2code 模型，它是一种给定 GUI 图片作为输入，且能生成计算机代码的新方法。虽然我们的工作展示了这样一种能自动生成 GUI 代码的潜力系统，但我们的工作只是开发了这种潜力的皮毛。我们的模型由相对较少的参数组成，并且只能在相对较小的数据集上训练。而构建更复杂的模型，并在更大的数据集上训练会显著地提升代码生成的质量。并且采用各种正则化方法和实现注意力机制（attention mechanism [1]）也能进一步提升生成代码的质量。同时该模型采用的独热编码（one-hot encoding）并不会提供任何符号间关系的信息，而采用 word2vec [12] 那样的词嵌入模型可能会有所好转，同时独热编码（one-hot encoding）也会限制可支持的符号数量。最后，由于近来生成对抗网络（GAN）在图片生成上有极其出色的表现，也许我们可以借助 GAN 及其思想由 GUI 图像生成计算机代码。