

TF.hub迁移学习

导师: GAUSS

目录

- 1/ TF.hub简介
- 2/ TF.hub使用
- 3/ 实战TF.hub

TF.hub简介



TF.hub简介

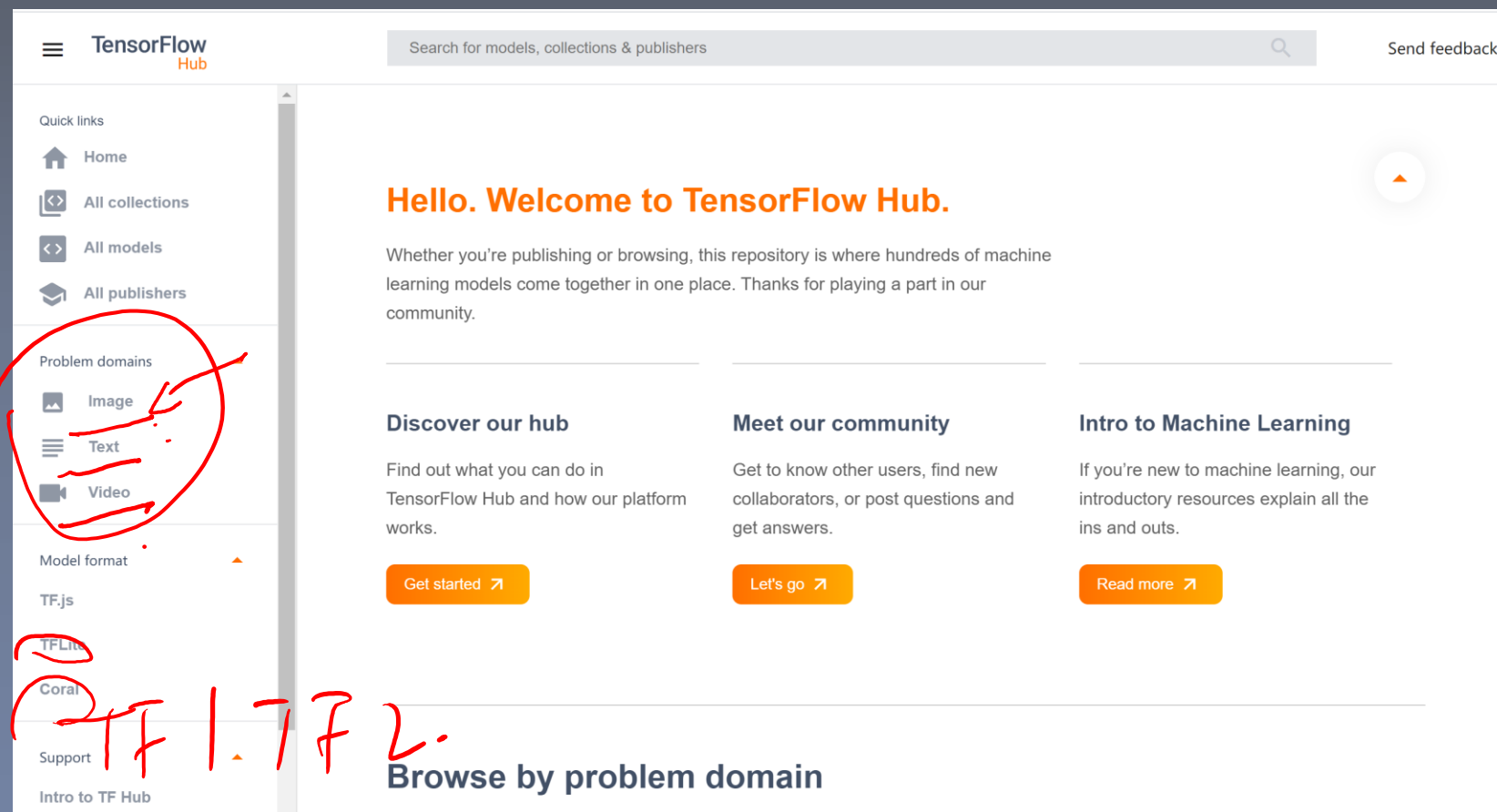
在软件开发中，我们经常复用开源软件或者库，避免了相同功能的代码重复开发，减少了大量的重复劳动，也有效缩短了软件开发周期。代码复用，对软件产业的蓬勃发展，有着极大的助推作用。

相应的，TF Hub 目的是为了为了更好的复用已训练好且经过充分验证的模型，可节省海量的训练时间和计算资源。这些预训练好的模型，可以进行直接部署，也可以进行迁移学习（Transfer Learning）。对个人开发者来说，TF Hub 是非常有意义的，他们可以快速复用像谷歌这样的大公司使用海量计算资源训练的模型，而他们个人去获取这些资源是很不现实的。

TF.hub简介

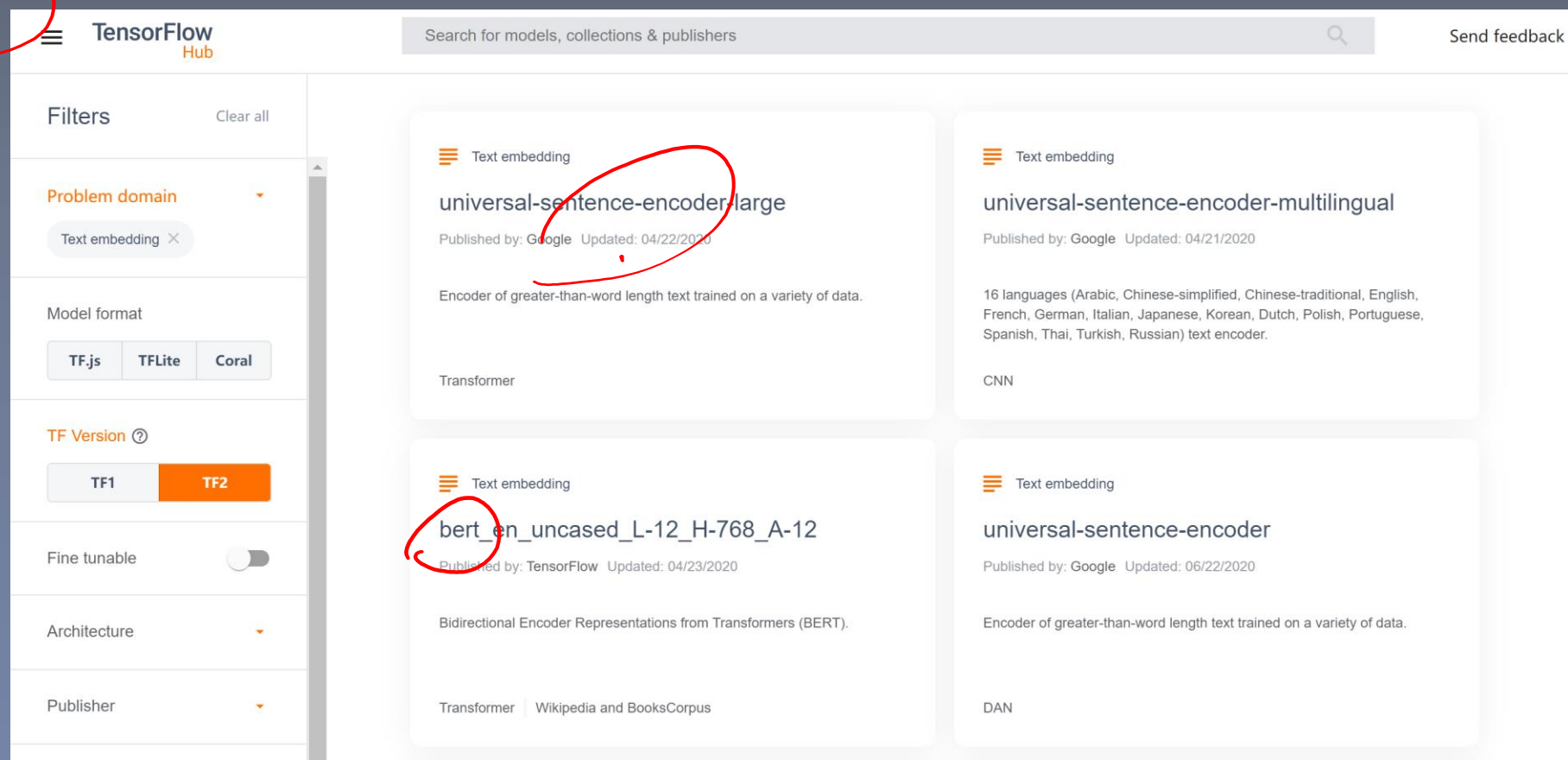
打开主页

<https://tfhub.dev/>，在左侧有 Text、Image、Video 和 Publishers 等选项，可以选取关注的类别，然后在顶部的搜索框输入关键字可以搜索模型。



TF.hub简介

以“Text embedding”为例：



The screenshot displays the TensorFlow Hub website with the search results for 'Text embedding' models. The left sidebar contains filters for 'Problem domain' (Text embedding), 'Model format' (TF.js, TFLite, Coral), 'TF Version' (TF1, TF2), 'Fine tunable', 'Architecture', and 'Publisher'. The main content area shows four model cards:

- universal-sentence-encoder-large**: Published by: Google, Updated: 04/22/2020. Encoder of greater-than-word length text trained on a variety of data. Transformer.
- universal-sentence-encoder-multilingual**: Published by: Google, Updated: 04/21/2020. 16 languages (Arabic, Chinese-simplified, Chinese-traditional, English, French, German, Italian, Japanese, Korean, Dutch, Polish, Portuguese, Spanish, Thai, Turkish, Russian) text encoder. CNN.
- bert_en_uncased_L-12_H-768_A-12**: Published by: TensorFlow, Updated: 04/23/2020. Bidirectional Encoder Representations from Transformers (BERT). Transformer | Wikipedia and BooksCorpus.
- universal-sentence-encoder**: Published by: Google, Updated: 06/22/2020. Encoder of greater-than-word length text trained on a variety of data. DAN.

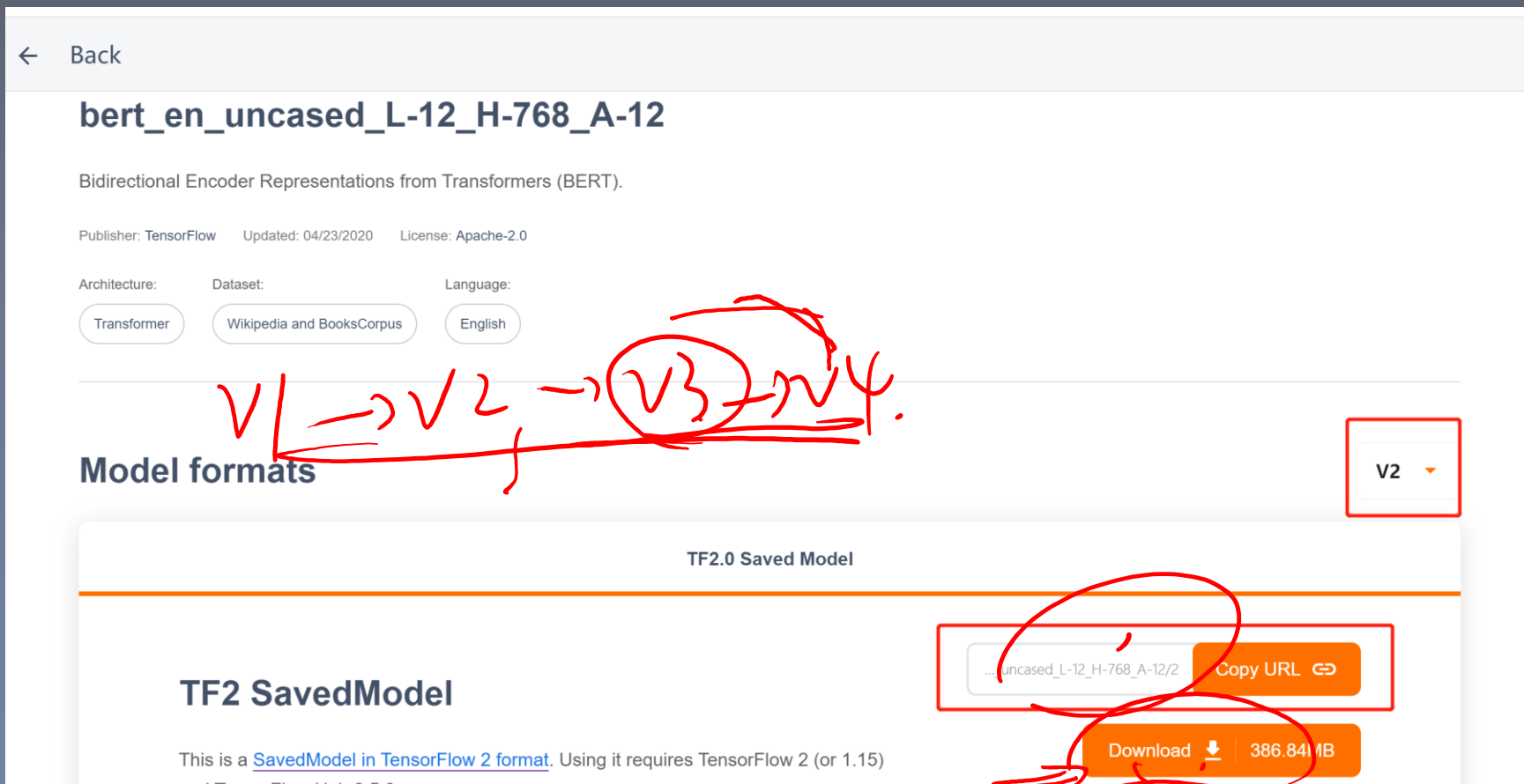
TF.hub简介

Bert模型

该模型的地址如下：

https://tfhub.dev/tensorflow/bert_en_uncased_L-12_H-768_A-12/2

其中，末尾的 2 为该模型的版本号。



← Back

bert_en_uncased_L-12_H-768_A-12

Bidirectional Encoder Representations from Transformers (BERT).

Publisher: TensorFlow Updated: 04/23/2020 License: Apache-2.0

Architecture: Dataset: Language:

Transformer Wikipedia and BooksCorpus English

Model formats

TF2.0 Saved Model

TF2 SavedModel

This is a [SavedModel in TensorFlow 2 format](#). Using it requires TensorFlow 2 (or 1.15) and TensorFlow Hub 0.5.0 or newer.

Download 386.84 MB

Copy URL

V2

V1 → V2 → V3 → V4.



TF.hub简介

注意目前还有很多模型是基于 TF1.0 的，选择的过程中请注意甄别，有些模型会明确写出来是试用哪个版本，或者，检查使用是否是 tfhub 0.5.0 或以上版本的 API hub.load(url)，在之前版本使用的是 hub.Module(url)。

如果不能访问 tfhub.dev，请大家转换域名到国内镜像

<https://hub.tensorflow.google.cn/>，模型下载地址也需要相应转换。

TF.hub使用



TF.hub安装

pip install tensorflow-hub

0.5.0

在 TF2.0 上，必须使用 0.5.0 或以上版本，因为接口有变动。



TF.hub接口

url 或 模型地址

```
hub.KerasLayer(  
    handle, trainable=False, arguments=None, _sentinel=None, tags=None,  
    signature=None, signature_outputs_as_dict=None, output_key=None,  
    output_shape=None, **kwargs  
)
```

Input

input_shape

handle: 一个可调用的对象

trainable: 一个布尔值, 控制该层是否可训练.

output_shape: 输出张量的形状

signature

符号

使用案例

- Bangla Article Classification With TF-Hub
- Exploring the TF-Hub CORD-19 Swivel Embeddings
- Retraining an Image Classifier
- Text Classification with Movie Reviews
- Transfer learning with TensorFlow Hub

图像再训练

作为第一个示例，让我们看一下一种可用于训练图像分类器的技术，该技术仅从少量训练数据开始。现在图像识别模型具有数百万个参数，当然，从头训练一个参数需要大量的标记数据和计算能力。使用称为“图像再训练”的技术，你可以使用更少的数据量和更少的计算时间来训练模型。

tf.hub

```
# Download and use NASNet feature vector module.  
module = hub.Module ("https://tfhub.dev/google/imagenet/nasnet_large/feature_vector/1")  
features = module(my_images)  
logits = tf.layers.dense(features, NUM_CLASSES) ← fe  
probabilities = tf.nn.softmax(logits)
```

> 1000

图像再训练

Hub

基本思想是重用现有的图像识别模块以从图像中提取特征，然后在这些之上训练新的分类器。正如你在上面看到的，在构建TensorFlow图时，可以从URL（或从文件系统路径）实例化TensorFlow Hub模块。TensorFlow Hub上有多种模块供您选择，包括各种版本的NASNet, MobileNet（包括其最新的V2），Inception, ResNet等。要使用模块，请导入TensorFlow Hub，然后将模块的URL复制/粘贴到你的代码中。

图像再训练



MobileNet

MobileNets come in various sizes controlled by a multiplier for the depth (number of features), and trained for various sizes of input images. See the module documentation for details.

- MobileNet V1

	224x224	192x192	160x160	128x128
100%	classification feature_vector	classification feature_vector	classification feature_vector	classification feature_vector
75%	classification feature_vector	classification feature_vector	classification feature_vector	classification feature_vector
50%	classification feature_vector	classification feature_vector	classification feature_vector	classification feature_vector
25%	classification feature_vector	classification feature_vector	classification feature_vector	classification feature_vector

图像再训练

每个模块都有一个已定义的接口，允许以可替换的方式使用它，而对内部的了解很少或根本不了解。在这种情况下，此模块具有可用于检索期望图像大小的方法。作为开发人员，您只需要提供一批形状正确的图像，然后在其上调用模块即可检索特征表示。该模块负责为您预处理图像，因此您可以在一个步骤中直接将一批图像直接转换为特征表示。从这里，您可以在这些基础之上学习线性模型或其他类型的分类器。

文本分类

让我们看第二个例子。想象一下，你想训练一个模型，将电影评论分为正面评论或负面评论，仅从少量训练数据开始（例如，数百个正面和负面电影评论的数量）。由于示例数量有限，因此你决定利用以前在更大的语料库上受过训练的单词嵌入数据集。

```
# Download a module and use it to retrieve word embeddings.  
embed = hub.Module("https://tfhub.dev/google/nnlm-en-dim50/1")  
embeddings = embed(["The movie was great!"])
```

文本分类

和以前一样，我们从选择模块开始。TensorFlow Hub提供了多种文本模块供你浏览，包括多种语言（EN，JP，DE和ES）的神经网络语言模型，以及受Wikipedia培训的Word2vec和受Google News培训的NNLM嵌入。

文本分类

NNLM embedding trained on Google News

Embedding from a neural network language model trained on Google News dataset.

	50 dimensions	128 dimensions
English	nnlm-en-dim50 nnlm-en-dim50-with-normalization	nnlm-en-dim128 nnlm-en-dim128-with-normalization
Japanese	nnlm-ja-dim50 nnlm-ja-dim50-with-normalization	nnlm-ja-dim128 nnlm-ja-dim128-with-normalization
German	nnlm-de-dim50 nnlm-de-dim50-with-normalization	nnlm-de-dim128 nnlm-de-dim128-with-normalization
Spanish	nnlm-es-dim50 nnlm-es-dim50-with-normalization	nnlm-es-dim128 nnlm-es-dim128-with-normalization

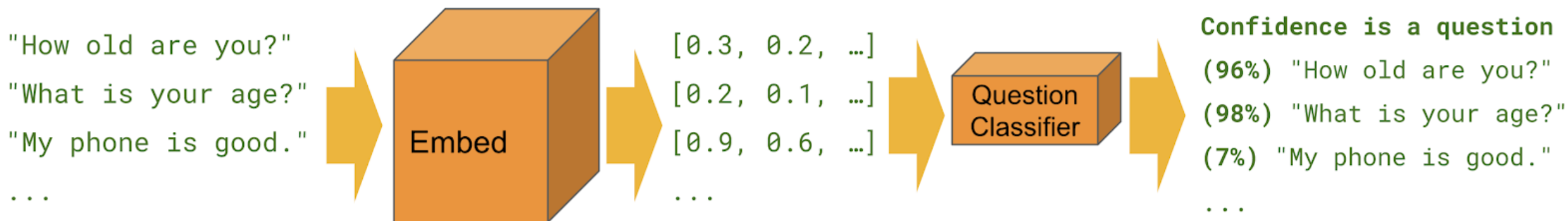


文本分类

在这种情况下，我们将使用一个模块进行单词嵌入。上面的代码下载了一个模块，使用它对句子进行预处理，然后为每个令牌检索嵌入。这意味着您可以在一个步骤中直接从数据集中的句子转换为适合分类器的格式。该模块负责将句子标记化，并处理其他逻辑，例如处理词汇外的单词。预处理逻辑和嵌入都封装在一个模块中，这使得更容易地试验单词嵌入的各种数据集或不同的预处理策略，而无需实质性地更改代码。

通用句子编码器

我们还共享了TensorFlow Hub模块的新功能！以下是使用通用句子编码器的示例。它是在多种数据集（换句话说，“通用”）上训练的句子级嵌入模块。它擅长的一些方面包括语义相似性，自定义文本分类和聚类。



通用句子编码器

与图像再训练一样，只需很少的标签数据就可以使模块适应您的任务。例如，让我们在餐厅评论中尝试一下。

```
# Use pre-trained universal sentence encoder to build text vector
review = hub.text_embedding_column( "review",
                                    "https://tfhub.dev/google/universal-sentence-encoder/1")
features = { "review": np.array(["this movie is a masterpiece", "this movie was terrible", ...]) }
labels = np.array([[1], [0], ...])
input_fn = tf.estimator.input.numpy_input_fn(features, labels, shuffle=True)
estimator = tf.estimator.DNNClassifier(hidden_units, [review])
estimator.train(input_fn, max_steps=100)
```


其他模组

TensorFlow Hub不仅仅是图像和文本分类。在网站上，您还将找到用于Progressive GAN和Google Landmarks Deep Local Features的几个模块。

实战TF.hub

实战TF.hub

下载: <https://tfhub.dev/google/tf2-preview/nli-zh-dim50/1>

总结

本节小结

Summary

TF.hub迁移学习	TF.hub简介	
	TF.hub使用	
	实战TF.hub	

结语

——我 说——

看过千万代码，不如实践一把！





深度之眼
deepshare.net

联系我们:

电话: 18001992849

邮箱: service@deepshare.net

QQ: 2677693114



公众号



客服微信

