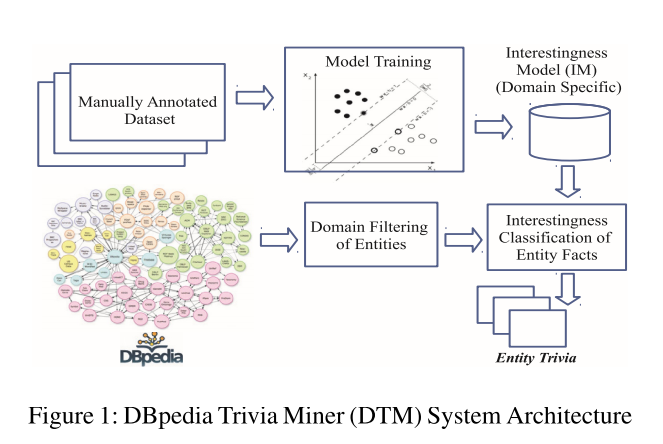
The Unusual Suspects:Deep Learning Based Mining of Interesting Entity Trivia from Knowledge Graphs

简介：该文章讲述了如何挖掘一个实体的有趣的事实（有趣是因为它的不寻常性、独特性、意外性），而这些事实能够促进用户去参与对于特点给定的实体的产品的体验。知识图谱是一个语义网络，这个语义网络是由关于实体和它们之间的关系的许多事实所编码。这篇文章提出了一个新奇的方法，被称作DBpedia Trivia Miner（DTM），对于在给定的知识图谱的范围的许多实体，它能自动的挖掘这些有趣的事实。DTM的关键在于学习一个兴趣模型（IM）。这篇文章提出了两个学习IM的不同方法，一个是CNN，另一个是F-CNN（Fusion-CNN）。其中F-CNN是将手动制作的特征与CNN的特征相结合。实验是在两个不同的域进行，一个是宝莱坞演员域，另一个是音乐艺术家域。最终DTM实现了F1 score对于两个不同的域分别达到了0.81与0.65。其中F-CNN相比于只使用手动制作的特征的基本方法的性能更好。

文章所使用的数据集在：<http://ltrc.iiit.ac.in/downloads/triviaminkg/trivia-mining-from-knowlegde-graphs.tgz、>

DTM的架构如下:



首先，DTM使用手动标注的训练数据（事实+标签，事实的格式是：实体，关系，客体，标签是二值的：boring或者是interesting）来学习一个指定域的兴趣模型。然后，对在给定域的每一个实体，这个兴趣模型被用来过滤出有趣的事实作为trivia。对于这个兴趣模型而言，我们将它看做是二分类问题（boring或者interesting）。有两个不同的方法去训练IM—CNN或者F-CNN。

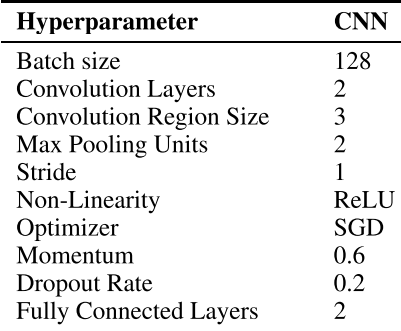
CNN for IM：

CNN使用一个知识图谱的事实作为输入（以三元组的格式：实体、客体、关系）。对于实体和客体而言，我们使用开源工具（http://github.com/idio/wiki2vec）生成300维的向量。这个工具是word2vec的微调版，这个版本不会将实体或者客体看做是分开的单词，而是将其看作是一个单独的单元。例如“The Load of the Ring”会被当做是一个单独的实体，将会有一个单独的向量。对于关系而言，我们使用word2vec生成300维的向量。调整关系的长度到8个单词，因为大部分的关系都比较短并且小于8个单词。具体的调整方法：如果小于8个单词，则将剩下的词向量置为0。如果大于8个单词，则将忽略剩下的单词。输入矩阵然后通过两组卷积、池化、非线性层以获得输入的三元组的300维的特征表示。使用最大池化操作为池化层，使用ReLU作为非线性层。这个300维的向量表示输入全连接神经网络。这些层建模学习的特征之间的各种相互作用，并最终输出概率得分，这个分数是表示给定事实的兴趣分数。

ReLU：线性整流函数（Rectified Linear Unit,ReLU）,一种激活函数。通常意义下，线性整流函数代指代数学中的斜坡函数，即：f(x)=max(0,max)。

>>综上：文章所使用的CNN方法的实现方法

1. 将实体与客体输入wiki2vec，生成300维的词向量。将关系进行截取，截取得到8个word的关系，将关系输入word2vec，生成8\*300维的词向量。
2. 按照如下方式构建CNN。



1. 将第1步中得到的词向量（（8+2）\*300）输入以上所构建的CNN，最终即可得到一个兴趣分数。

>>现在存在的问题：

1. 对于wiki2vec在github上给出了在ubuntu14.04上的配置方法，因此我打算将其配置在ubuntu14.04上。目前虚拟机已经安装完成，但是网络存在问题，连接不上ubuntu的源，并且登不上github。
2. CNN的构建存在问题。没有使用过python构建神经网络，对于这块还不太清楚，不知道怎么弄。

Hand Crafted Features（HCF）

给定一个事实三元组（实体，关系，客体），可以计算许多的可以帮助我们抓住一个事实的不寻常性的特征。

谓语（关系）特征（Predicate Features）：

逆实体频率（IEF）：在信息检索（IR）中，逆文件频率（IDF）通常是用来测量术语的重要性。此处的逆实体频率与之类似。对于一个给定的关系/谓语p，可以找到它是在给定域D里的稀少程度。我们相信，属于同一个域里的实体将有许多相同的谓语，因此少有出现的谓语将可能是有趣的。我们在KG中定义与逆文件频率类似的逆实体频率。对于给定的谓语p，有如下的逆实体频率定义：



其中，|D|表示定义域中所有的实体的数量，np表示所有实体中谓语p出现的次数。例如，对于宝莱坞演员域，一个谓语（比如starring）是几乎所有的实体中都有出现，因此它有很低的IEF值。然而一个谓语（author）将会有很高的IEF值，因为只有少数的宝莱坞演员才出版过书籍。

谓语频率-逆实体频率（PF-IEF）：另一个在信息检索中非常著名的术语是TF-IDF。我们在KG中定义与之类似的概念。对于一个实体e和一个谓语p：



PF(p,e)=谓语p发生的实体e的RDF三元组数。

谓语词袋（Predicate bag of words）：这个做法是将谓语的文本转化成一个词袋，然后从中提取特征。一些谓语可能比其他的谓语更有趣。例如，owner，founder，leader。（不太清楚谓语词袋在这里有什么用，也并未给出一个计算方法）

（Bag of Words模型：也叫做“词袋”，在信息检索中，Bag of words model假定对于一个文本，忽略其词序和语法，句法，将其仅仅看做是一个词集合，或者说是词的一个组合，文本中每个词的出现都是独立的，不依赖于其他词是否出现，或者说当这篇文章的作者在任意一个位置选择一个词汇都不受前面句子的影响而独立选择的。）

语义类别特征（Semantic Category Features）：

Object Type：给一个RDF三元组，我们发现object o的语义类型例如Person，Place，Organization，Work，EducationalInstitution，File等等。在一些谓语中这可能是有用的特征用来得到不寻常的事实。（并未给出具体的计算方法）

Object Categories：（并未给出具体的计算方法）

Average Category and Type Entropy：

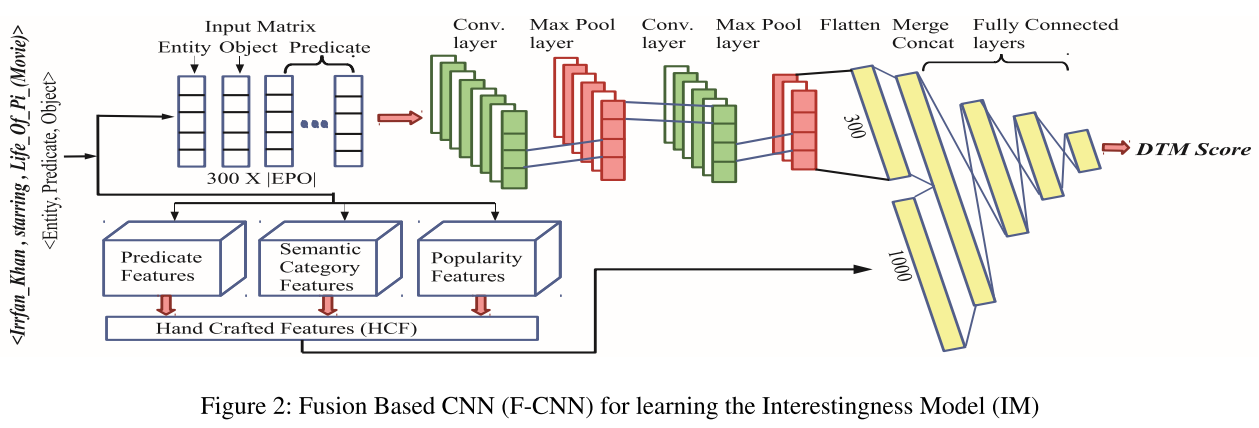


其中Pr(c|M)是整个域M中的category/type标签c的概率。如果有许多与对象相关联的稀有类别，那么E就会非常高。

Popularity：更受欢迎的实体的trivia会产生更大的interesting。一个网页的点击次数通常表明它的受欢迎程度。例如，一个三元组<Michael Jackson，owner，Bubbles(chimpanzee)>。因为mj是非常受欢迎的，所有人们可能会发现与他有关的不寻常的事实是非常有趣的。因此对于每一个RDF三元组，我们可以利用他们的维基百科的主页的点击次数去估计实体e和客体o的相对受欢迎程度。我们使用the Mediawiki Awk API去计算维基百科的点击次数。（http://github.com/greencardamom/MediaWikiAwkAPI）

F-CNN for IM:

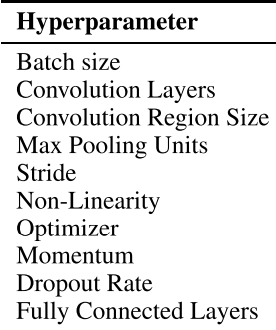
F-CNN的架构如下：



F-CNN在第一步和CNN是相似的。将KG fact作为输入与CNN的格式相同。然后输入的事实通过两个卷积层、池化层和非线性层生成了一个300维的特征表示（将其称作CNN-FR）。与此同时，使用输入的事实，F-CNN像之前描述那样计算hand-crafted features（将其称作HC-FR）。第二步，将两种特征结合起来，然后输入全连接神经网络，在这个全连接神经网络中最后的输出节点给出了目标类的概率。为了避免大量的没有意义的HCF features通过全连接神经网络，我们进行了基础特征选择并且只选了其中含有信息量较大的前1000个特征。对于基础特征选取，我们选择正则化嵌入模型方法，在其中我们训练一个使用L1正则化的线性SVM（SVM-L）根据各个特征的重要程度进行排序选择其中前1000的特征。对于每一个域，对于特征选取的SVM的训练是从对应的数据集中随机使用5000个事实样本进行的。

>>综上：文章所使用的F-CNN的实现方式如下

1. 与CNN类似，同样提取出300维的特征出来。F-CNN按照如下进行配置：



1. 提取hand-crafted features。利用SVM-L选取主要的1000个HCF。
2. 将CNN所提取的特征与hand-crafted特征相结合。然后将其输入最后的全连接神经网络，最终输出节点输出最后的目标类的概率。（还记得前面是将其当做是一个二分类问题，boring or interesting）。

>>现在存在的问题：

1. 不知道如何获取hand-crafted features。（虽然前面介绍了一大堆和它相关的）

在hand-crafted features中，有Predicate Features，Semantic Category Features，Popularity，但是这些应该怎么组合起来。（似乎这个问题就是使用文章所说的SVM-L来进行选择，选取比较重要的1000个特征。但是似乎按照前面所说的Predicate Features，Semantic Category Features，Popularity的计算一共就得不到1000维的特征）。

1. 待续。