基于循环神经网络的谣言监测

# 一、文章内容

1.摘要：微博平台是一个散布谣言的理想平台，自动揭露谣言是一个至关重要的问题。为了检测谣言，现存的方法都是依赖于手工提取特征，来应用于机器学习算法，这需要大量的人力。当遇到一个有争议的言论，人们需经过一段时间，列举多样的线索，来争论它的真伪性。这样会产生对证据的长距离依赖性。这篇论文，提出了一个新方法，通过学习连续的微博事件表征来识别谣言。提出的模型是基于学习隐藏表征的循环神经网络，来捕获相关推送在一段时间内的上下文信息的变化。实验结果是基于来自两个真实世界的微博平台的数据集，结果表明：（1）RNN方法比使用手工提取特征的谣言检测模型好。（2）以RNN为基础的算法表现通过复杂的循环单元和额外的隐藏层被进一步提高。（3）以RNN为基础的方法比现存的技术，包括领先的在线谣言揭露服务更快，更准确。

2.介绍：社会心理学文献将谣言定义为一个真值未经证实或故意为假的故事或状态。假的谣言是有害的，因为其引起公众的恐慌和社会的不安定。比如，在2015年8月25日，一个内容为“韦拉克鲁斯的学校附近正在发生贩毒团伙枪战和绑架”的谣言通过Twitter和Facebook传播。这引起了26起车祸，引发了城市的严重混乱，因为人们纷纷将自己的车扔在马路中央，奔跑去接自己在学校的孩子。这场假新闻引发的事故强调了自动预测社交媒体上信息的真实性有着重要的实际意义。

在谣言传播的早期将其揭穿对于最小化他们的有害影响有着尤其重要的作用。为了将谣言从真实事件中区分出来，个体和组织往往依赖于常识和新闻调查。像snope.com和factcheck00.org谣言报道网站正在共同努力。但是，由于人工验证步骤需要大量精力，这些网站的话题覆盖范围不够综合，并且也有长时间的揭穿延迟。

现存的谣言检测模型使用的学习算法是融合大量特征，这些特征是从推送的内容、用户特征和扩散模式这些方面手工提取的，或者简单利用了常规表达中的表达特征来发现推特中的谣言。特征工程是至关重要的，但也是精心雕琢的细节、有偏见的、耗费大量劳动的。比如，表1中的两个时间序列图描绘了谣言信号典型的浅层模式。虽然它们可以揭示谣言和非谣言事件的时间特征，但是对于特征工程来说，这种区别不够明确也不够有力度。

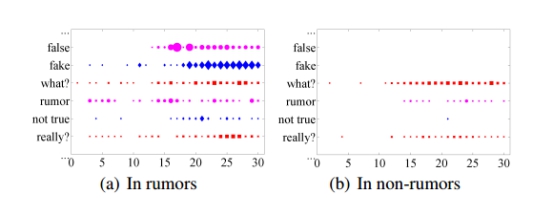


图1：一些伴随事件时间轴能标记谣言的浅层模式（从最初推特发布后的小时数内），图形的大小表示相关频率的强度

从另一方面来说，深度神经网络对很多机器学习问题表现出了明显的优势。在这个研究中，我们发现了自动发现谣言的可能性并且挖掘了对有效谣言检测的深度数据表示。我们假定给出的社交媒体上的文本流是顺序特征的，那循环神经网络是适用于谣言检测的。这是因为RNN中单元之间的联系构成一个直接的循环，并创造了一个网络的内部状态，这样能够捕捉到谣言传播的动态时间信号特征。

利用RNN，我们将一个事件的社交文本信息建模为一个可变长度的时间序列。我们假设当人们接触一个谣言时，会转发或者评论，这样就建立了一个连续的信息流。这种方法同时从谣言帖中学习时间和文本特征，是监督学习。通过在两个现实世界的微博数据集上的大量实验，表明基于RNN的模型表现优越。这个模型同样被证明对于早期的谣言探测是有效的，在这个模型下，在谣言初次发布的几个小时内即可获得足够的准确性。

这篇文章的贡献有下面三个方面：

\*尽我们所能，这是第一个将深度学习模型应用于微博谣言检测的研究。基于RNN的模型对比依赖于手工提取特征的学习算法，取得了显著的提高。这个模型更丰富并且可以通过使用复杂的循环单元和额外隐层使谣言检测更加准确。

\*基于RNN的模型允许早期检测，在与现存的标准，比如领先的网络谣言揭露服务相比较时，显示出明显的功效。

\*我们为这个实验构造了两个带有真实标签的微博数据集，总共包含超过5000个消息，涉及约五百万个相关的微博帖子。我们公开了这个大型的谣言数据集，公众可以用来做研究。

3.相关工作

社交媒体的自动谣言检测以传统的分类器为基础，这些传统的分类器检测谣言源于以往在推特上的信息可信度的研究[Castillo et al., 2011]。在接下来的工作中，[Yang et al.,2012; Liu et al., 2015; Ma et al., 2015; Wu et al., 2015]不同的手工提取特征集合被提出并合并来决定关于一个事件的消息是否是可信的。大部分这些以往的工作试图使用信息而不是文本内容来对传播的模因进行准确性分类，举例来说，一个帖子的人气（例如：帖子的转发、回复量）、有关决定用户信誉度的特征等。但是，特征工程需要大量精力。我们的基于RNN的方法完全不考虑这个，但可获得更好的表现，这是由于深度神经模型有效的特征表现学习能力。

以往的一些研究聚焦于在谣言的传播过程中捕捉到其时间特征。Kwon引进了一个基于单个特征时间属性的时间序列拟合模型---推特量。Ma使用动态时间序列捕捉一系列社交语境的变化来拓展了这个模型。Friggeri通过分析与谣言揭露网站有连接的评论，阐释了Facebook上错误信息级联的结构。我们也使用了表示的时间性质，但是这些特征通过在一个在每个时间段中给出基本术语表示的RNN被自动学习。

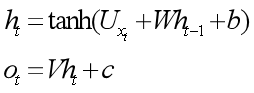
Zhao，致力于早期谣言检测，通过使用例如“不是真的”、“未经证实”或“揭露”等提示词来发现具有疑问和错误的推特。我们的RNN模型学习到的表现显然比这些单一的信号更加复杂；这些表示可以捕捉隐藏的含义和随时间的依赖关系。

我们的工作也与以下研究相关联，又有不同。包括“Lee的检测垃圾信息制造者、Gupta的Twitter上的虚假图片、Ratkiewicz的用来区分一个模因是否被有机还是人工传播的“Truthy”系统。

4.基础知识

（1）RNN：循环神经网络

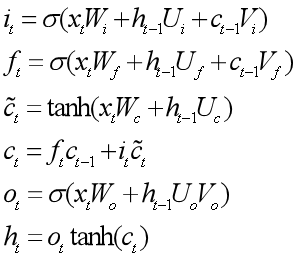
RNN是一种前馈神经网络，它可以被用于像句子或时间序列这样的变长连续信息建模。一个基本的RNN形式如下：给定一个输入序列（x1,...,xT)，在每个时间步长，模型更新隐含层状态（h1,...,hT），并生成输出向量（o1,...,oT），T取决于输入长度。从t=1到T，算法通过下面的等式迭代：



通常，RNN的梯度经过时间的反向传播来计算。实际上，基础的RNN由于梯度消失问题，不能用基于梯度的最佳化算法来学习长距离的时间依赖。来解决这个问题的一个方法是做一个扩展，包括“记忆”单元来存储长时间的信息，通常称为LSTM和GRU。

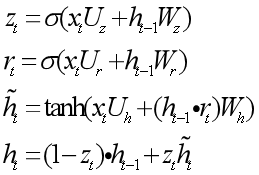
（2）LSTM（长短时序循环神经网络）

不同于传统的循环单元，每一时间步的状态都被重写（公式1），LSTM在时间t有一个细胞单元Ct。单元在t时刻保持一个存储单元ct。LSTM的输出ht通过下列等式计算：



（3）GRU

同LSTM类似，GRU由门控单元来调节单元内部内容的流动。但GRU比LSTM有较少的参数。下面等式用来描述一个GRU层。



4.模型实现

我们展示了我们将微博事件分成谣言或非谣言的基于RNN的模型的细节。首先，我们介绍了一种方法，它将微博帖子的传入流转换成连续的可变长度的时间序列，然后描述RNN用以进行分类的不同种类的隐含单元和隐含层。

1. 问题陈述

单个微博帖子篇幅短，包含有限的文本内容。一个声明往往与大量同这个声明相关的帖子相联系。我们对集合层感兴趣而不是单层。因此，预测每个帖子的真实性并不是我们的焦点。我们集中于在事件层级上进行谣言检测，事件由相关帖子的集合组成。

1. 可变长的时间序列

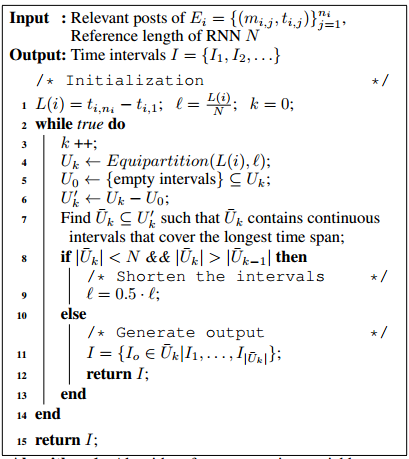
我们可以将每个帖子作为一个输入实例来建模，并建立一个RNN模型，其时间序列的序列长度与帖子数相同。但是，在一个热门事件中可能会有成百上千的帖子。我们只有一个在每个事件最后的时间步长显示分类的输出单元。在反向传播时，要通过大量时间步，却只有一个最终的损失，它的计算代价将会很高昂且效果不理想。因此，我们在时间间隔内，批处理帖子，并将它们看作一个时间序列中的单独单元，之后使用一个RNN序列建模。时间序列长度，采用RNN的参考长度。

我们定义一系列给定事件的集合E={Ei}，每个事件Ei在理想情况下包含所有在时间戳tij下的与之相关的帖子mi，j，Ei={（mi，j，ti，j）}任务是分类一个事件是否是谣言。

1. 变长时间序列

扩散中密集帖子的时间跨度应该被正确捕获；时间间隔的数量应采用接近RNN的推荐长度。算法1描述了步骤。

首先，将整个时间线等分成N份（N是推荐长度），然后，我们的系统试图发现非空间隔集合U’(举例，U’中的每个间隔，有至少一个推送）通过将空间隔移入U0，U’中连续间隔的时间跨度最长的被选入U,如果U中间隔数量少于N，并且间隔数量比上一轮数量多，我们二等分间隔并且继续分割。否则，返回U中发现的连续间隔，标记整个时间序列长度，接近于N，但不同的事件长度不同，然而一个事件中的单个间隔相同。

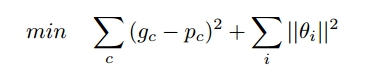


算法1：算法用来在给定事件的一系列相关帖子和RNN的推荐长度时构造变长时间序列

1. 模型结构

基前文构造的时间序列，RNN模型适用。在每个间隔，我们使用tf\*idf方法提取的词汇作为输入。我们根据它们的tf\*idf值，通过保留top-K项来修剪词汇表，所以输入维数是K.我们将RNN拓展为三个结构，如图2所示。注意，输出单元和最后的时间步有关，使用softmax来表示两类的概率输出。tanh-RNN 是一个基础结构，不包括门控单元。因此，只能有限的获取时间间隔内的文本。

gc ，c表示类标签，是一个事件的二维多项式分布的真实输出。这里，定义分布为，[1,0]是谣言,[0,1]不是。对每个训练集（例如，每个事件），我们的目标是缩小估计值和真实值之间差的平方。

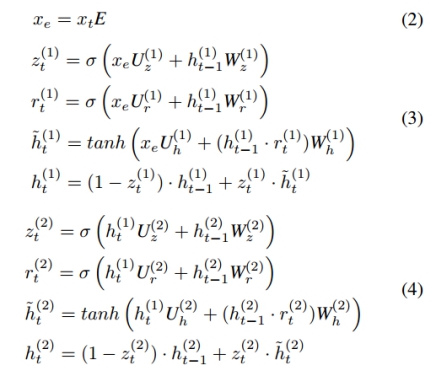


gc 和 pc 分别是真实和估计值，θi是模型的估计参数，L2正则化惩罚被用于替换错误和问题规模（防止过拟合）。Tanh-RNN如图2（a）

单层LSTM和GRU。长距离依赖在事件的生命周期中，对于捕捉谣言的特征和隐藏信号十分重要。我们使用LSTM和GRU调整了循环单元为门控单元。RNN的门控单元如图2（b）所示。门控单元不仅保留了当前时间步的内容，并且从以往的时间步中注入了相互依赖的证据。

但是，因为门控单元，参数的规模急剧扩大。举个例子，GRU因为引入重置门和更新门，参数变为以往的三倍。为了减少复杂性，我们在输入和隐藏层之间加入嵌入层（固定长度为100），这样整个参数规模变得很小。包括嵌入层 将单词向量转变为低维度。不同于使用基于外部收集的预训练过的向量，我们使用我们的模型自主学习嵌入矩阵E。

多层GRU.我们进一步发展单层GRU，通过加入第二层GRU，来捕获不同时间步之间更高层的特征交互。考虑到LSTM参数的复杂，我们只扩展GRU为多层。图2（c）展示了结构。隐藏层的值通过使用等式2-4收集，嵌入层由等式2给出，第一层和第二层GRU层由等式3和等式4给出。



E是词嵌入权重矩阵。（U(1),W(1)）和（U(2),W(2)）是GRU中第一层和第二层隐藏层的权重关系。

模型训练。 我们通过使用关于所有参数的反向传播的损失导数来训练所有的RNN模型。我们使用了AdaGrad算法来更新参数。我们设置词汇大小K为5000.embedding 大小为100，隐藏单元的大小为100，学习率为0.5.我们在每个阶段迭代所有训练事件并且持续直到损失收敛或者最大纪元数满足。

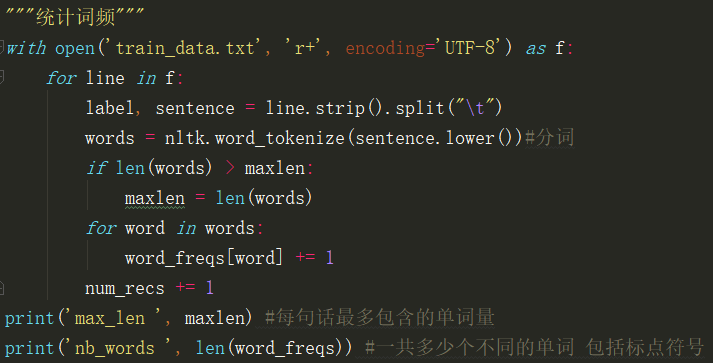
# 二、实验过程

1、数据集

数据来自Kaggle的情感分类比赛（https://www.kaggle.com/c/si650winter11）。一共7086条评价。正面评价标记为1，负面评价标记为0。

2、代码实现

（1）统计词频

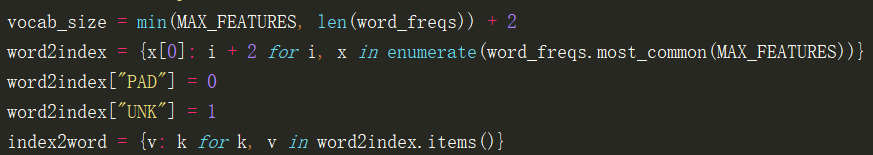


（2）准备词汇表

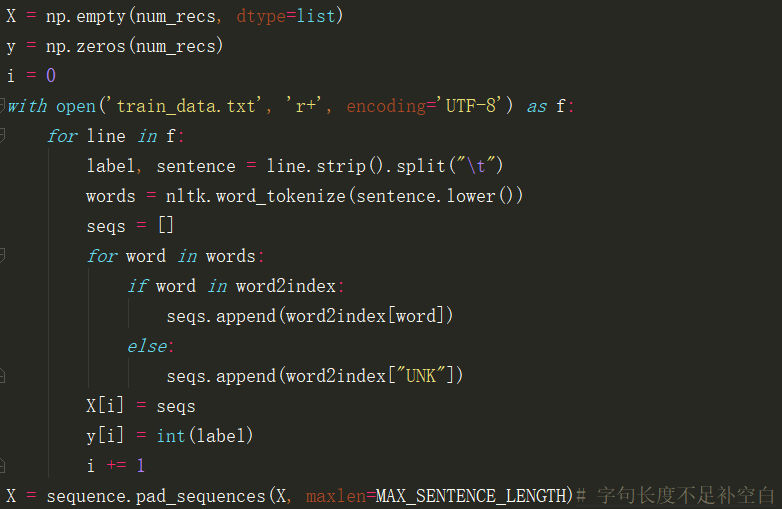
根据句子最多包含的单词的个数，我们可以把词汇表的大小设为一个定值，并且对于不在词汇表里的单词，把它们用伪单词UNK代替。根据句子的最大长度，我们可以统一句子的长度，把短句用0填充。

根据第（1）步统计词频结果，已知：句子的最大长度为42，包含的单词总共有2330个。所以，我们把VOCABULARY\_SIZE设为2002。包含训练数据中按词频从大到小排序后的前2000个单词，外加1个伪单词UNK和填充单词0。最大句子长度MAX\_SENTENCE\_LENGTH设为40。

接下来建立两个lookup tables，分别是word2index和index2word，用于单词和数字转换：



根据lookup table把句子转换成数字序列，并把长度统一到MAX\_SENTENCE\_LENGTH，不够的填0，多出的截掉：

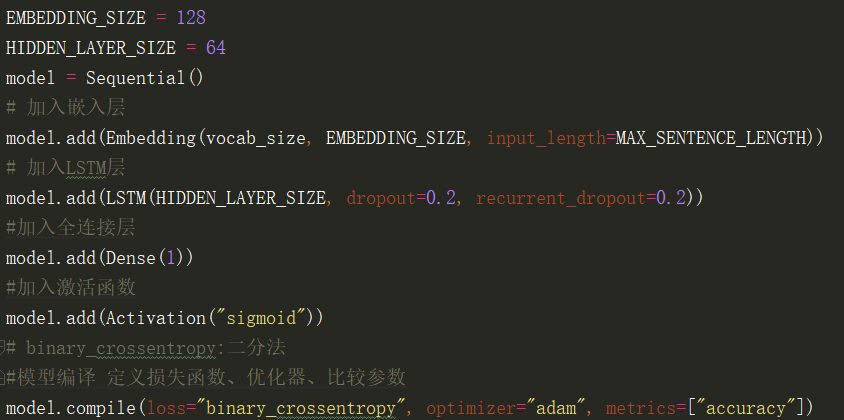


（3）模型定义

首先划分训练集和测试集：80%作为训练数据，20%作为测试数据

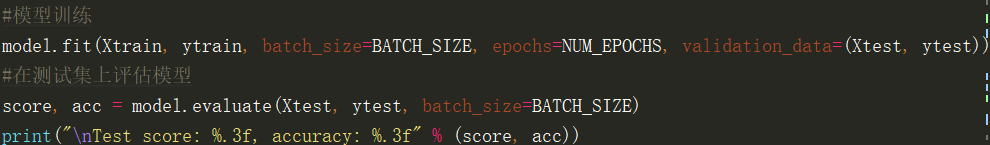


然后根据论文所述结构，加入嵌入层、隐藏层（LSTM层）、全连接层、激活函数。定义损失函数、优化器和比较参数。

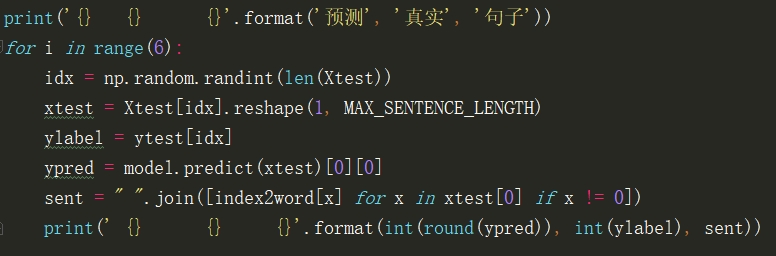


（4）模型训练及测试

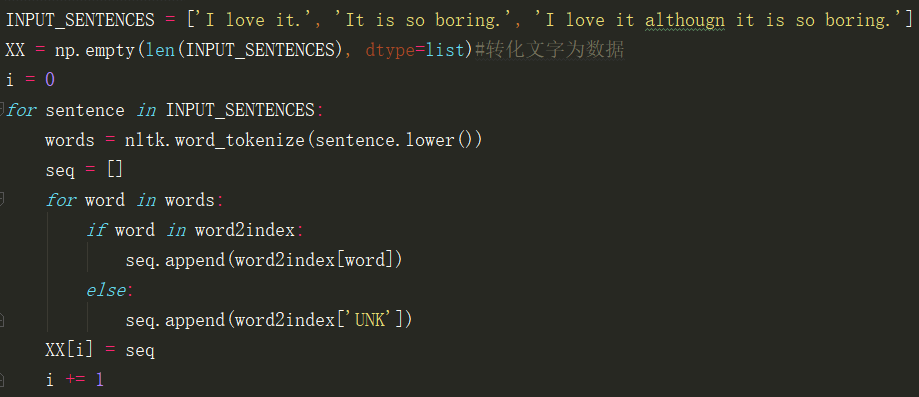
Epochs取10，batch\_size取32，训练模型。并用划分好的测试集来评估模型的准确率。

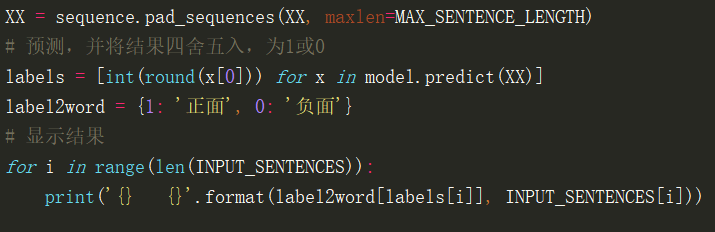


用已经训练好的模型随机选取6个测试集的数据进行测试。



自己输入评价语言测试结果。



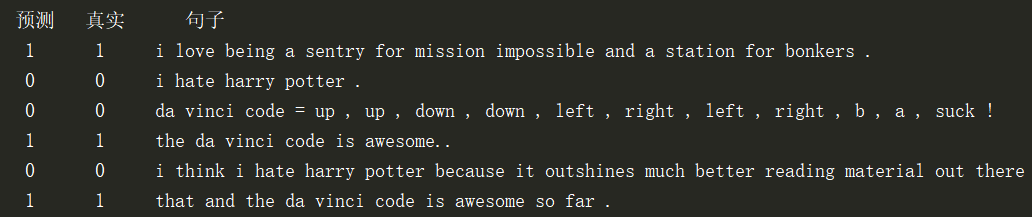


三、实验结果

测试集上评估分数以及准确率：



随机选取6个测试集评论运行模型：



自己输入的三个评论的正负面判定：

