1. 数据预处理

1.1. 主要参数

主要参数	取值
采样周期（sample cycle）	6H
观测窗长（observation window length）	7
预测窗长（prediction window length）	1
最低流行度（min popularity）	5

1.2. 数据格式

数据是2016年10月3号到10月30号某小区对youku和iqiyi视频网站的访问记录，数据格式为三元组的形式（用户，时间，视频ID）。通过记录每个视频在采样周期内的流行度，我们可以把数据转换为流行度随时间变化的时序数据。然后筛选流行度高于最低流行度的视频，并且按预测窗长和观测窗长对数据进行切分，从而整理成可供算法学习的格式。数据基本统计情况如下：

	youku	iqiyi
请求数		
用户数		
视频数		

1.3. 训练集测试集划分

由于请求数据有着时间标签，因此测试数据不能出现在训练数据之前。所以我们需要根据时间来对训练集和测试集进行划分。鉴于此，我们将前三周作为训练集，最后一周作为测试集。验证集从训练集中随机抽取。

1.4. 数据归一化

数据归一化是数据预处理的重要组成部分，合适的归一化手段可以加速模型收敛。参考论文[1]的归一化探究，Tanh和Sigmoid的归一化效果比较好。由于Z-score是最常用的归一化方法，因此我们选择这三种归一化方法进行比较。

- Z-score
- Tanh
- Sigmoid

2. 特征工程

对于流行度预测而言，历史的流行度变化可能是最关键的特征。此外一些附加信息也能起到预测的效果，如辅助时间序列、文本特征等等。

2.1. 历史流行度序列

历史流行度序列的长度根据观测窗长来确定。对于该时间序列特征我们可以使用一阶差分以及Min，Max，Sum等操作来得到一些额外特征。

2.2. 相关时间序列

实际上影响视频流行度的因素可能有很多，单纯的历史流行度序列无法完全预测未来的流行度，一般可以寻找一些相关的时间序列来辅助预测。由于数据集包含同一批用户对youku视频和iqiyi视频的观看记录，因此可以使用基于用户的协同过滤找出与目标youku视频相似的iqiyi视频，从而相似视频的流行度序列即可作为相关时间序列来看待。论文[2][3]对于这种特征给出了SOTA的方案。

2.3. 时间位置特征

由于所有的视频流行度预测共用同一个模型，在某些情况下我们需要时间位置特征来对样本进行区分。比如在不同的日期，视频A在观测窗里的时间序列相同，通过模型后的预测值就一样，这显然是不合理的。因此我们需要加入额外的时间特征，如预测窗处于的时段、星期几、是否假期、距离该视频第一次被看的时长等等。

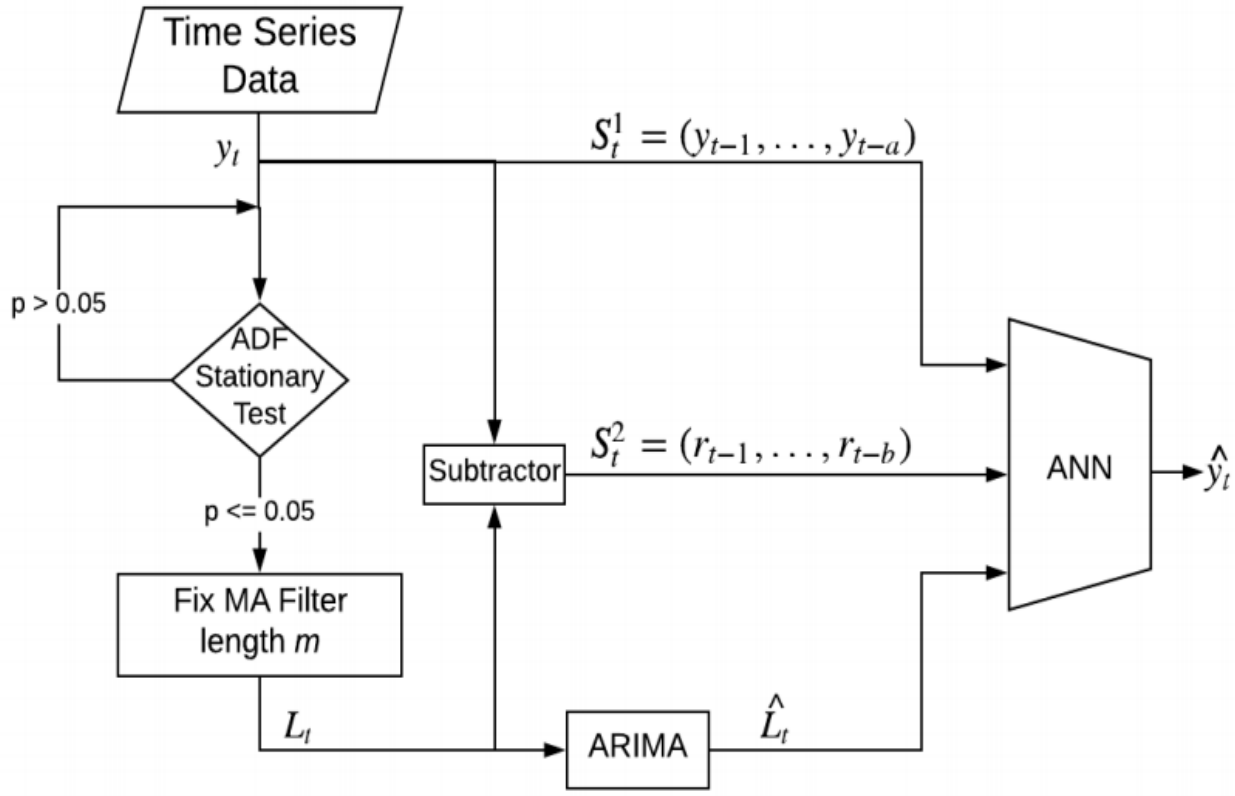
2.4. 文本特征（需要给一些相关的引用）

如时间位置特征一样，文本特征也能帮助我们对样本进行区分。考虑同一时间段的不同视频，它们在观测窗里的流行度序列一样，从而通过模型后的预测值也一样，这同样是不合理的。因此需要引入关于视频自身的一些特征，如视频标题、视频标签、视频描述等等。同时文本特征也可以缓解冷启动视频的影响。

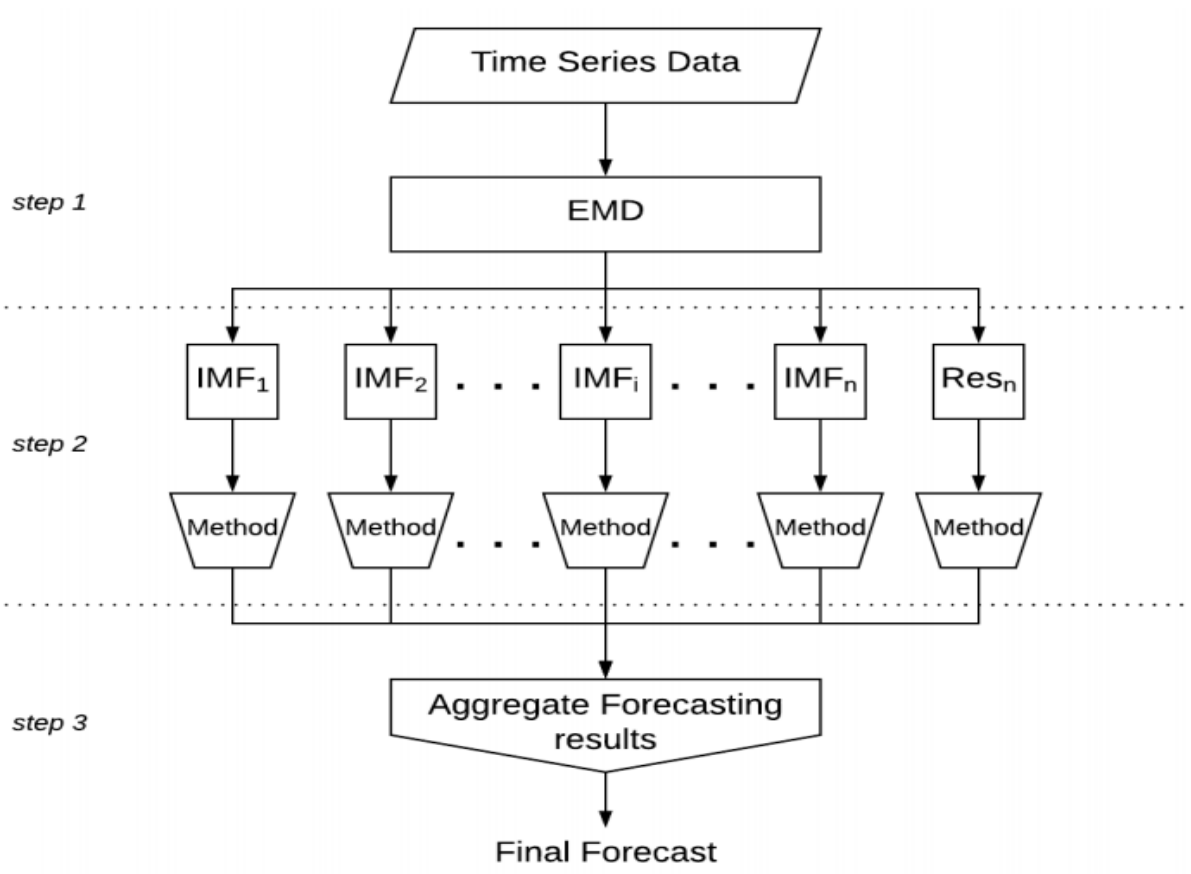
2.5. 时间序列分解

非平稳时间序列的预测相对平稳时间序列的预测难度更高，直接使用原始时间序列可能难以得到满意的结果。常用的分解方式有LN分解[4]和EMD分解[5]。

- LN分解：LN分解即是将时间序列分解成线性部分和非线性部分。论文[4]的实现方式是先使用线性模型ARIMA对时间序列进行拟合，然后使用ANN对残差部分进行预测，两个模型的加和构成最终的预测结果。



- EMD分解：将不稳定的时间序列分解成一系列更稳定的时间序列，即IMF序列。这些子系列有两个重要的属性，可以很容易地建模：（1）每个子系列都有自己的局部特征时间尺度（2）它们是相对固定的子系列。



3. 模型结构

由于数据的独特之处主要在于拥有两个不同视频网站的观看记录，因此模型主要聚焦在这个部分。

3.1. Wide-Deep架构

Wide-Deep架构最开始出现在论文[6]中，被作为推荐系统的架构提出，但之后渐渐发展成一个通用的深度学习框架。我们知道，深度模型如DNN，RNN等擅长提取高阶特征，而浅层模型如LR等擅长提取低阶特征，因此深度模型和浅层模型可以形成一个互补的Wide-Deep模型。此外论文[7]也指出，线性模型如AR，ARIMA等适合预测线性时间序列，而ANN等深度模型适合预测非线性时间序列。这也是我们采取Wide-Deep架构的原因之一，即浅层模型用来预测序列的线性变化部分，深度模型用来预测序列的非线性变化部分。不同于论文[7]中线性模型和深度模型分开训练的方式，Wide-Deep框架的联合训练方式或许能取得更好的效果。此外Wide-Deep也是端到端的模型，相比于论文[7]中的混合模型，训练和预测都更加方便。

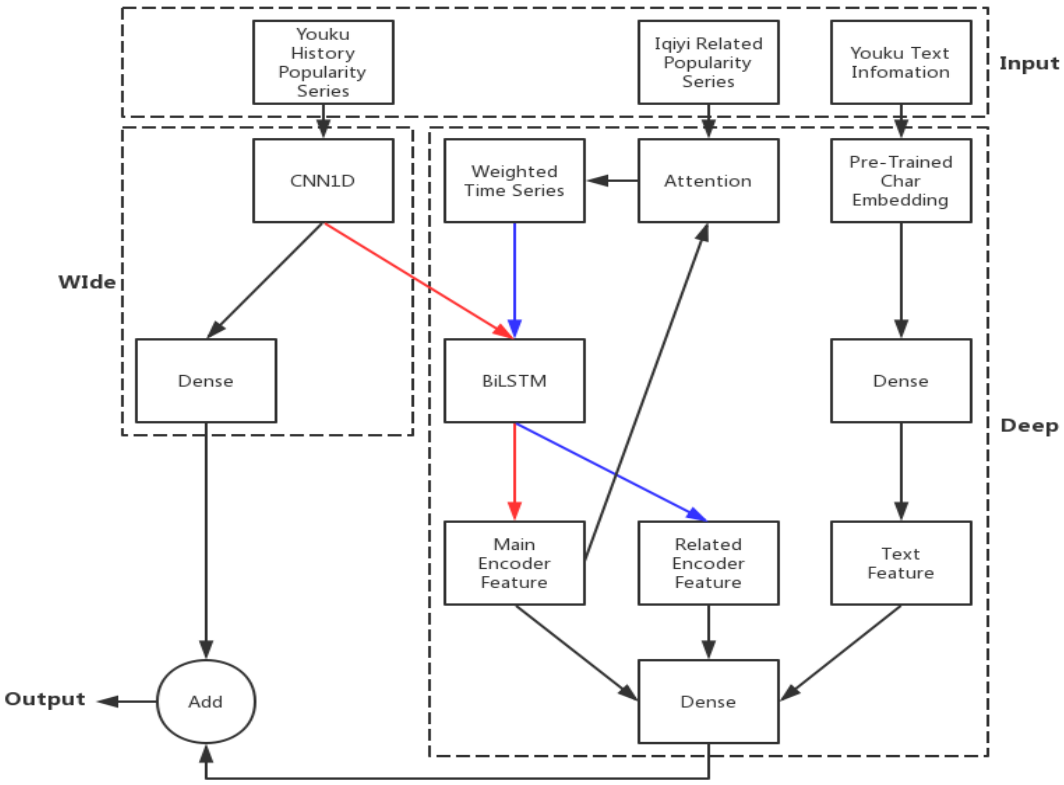
3.2. CNN/RNN/DNN

- CNN：类似于Moving Average的思想，我们使用一维CNN模块来对输入的时间序列特征进行平滑。相比于Moving Average的数据预处理模式，CNN模块不仅可以训练出一个局部加权的参数，而且可以使得整个模型是端到端的。
- RNN：RNN是对时序数据最优的处理方式之一，同时对于RNN的所有隐层状态，同样可以使用Attention机制来组合。此外传统RNN在训练时面临梯度消失的问题，我们使用长短期记忆网络LSTM来替代传统的RNN。
- DNN：DNN主要用作对文本嵌入之后较高维的特征作压缩。

3.3. Attention（还可以展开）

Attention主要作为一个特征选择的组件出现。在原始论文[8]中,作者使用Attention机制在解码器的不同时间步给编码器的所有状态赋予不同的权重，这种灵活的特征选择方式大大提升了机器翻译的性能。同样的，对于相关时间序列，可以根据目标时间序列的变化趋势来给它们赋予不同的权重。论文[2]采取Input Attention和dencoder-encoder Attention的模式对特征进行选择，取得了当前最优的性能。而相对于论文[2]中固定的相关时间序列，本文将针对不同的观测窗来选取不同的相关时间序列，这对于视频流行度快速变化的情况应该适应的更好。

3.4. 模型整体结构



3.5. 模型超参数

模型主要参数	默认值
LSTM隐藏层维数	8
DNN输出层维数	2
Attention模式	乘法Attention
LSTM和DNN层是否共享参数	True
是否加Wide部分	True
是否加CNN	False

4. 实验结果和分析

4.1. 数据归一化方式的影响

4.2. Attention的有效性

4.3. 模型超参数的影响

4.4. 特征重要程度分析

5. 参考文献

[1]Impact of Data Normalization on Deep Neural Network for Time Series Forecasting.
[2]A Dual-Stage Attention-Based Recurrent Neural Network for Time Series Prediction.
[3]Temporal Pattern Attention for Multivariate Time Series Forecasting.
[4]Time Series Forecasting Using a Hybrid ARIMA and Neural Network Model.
[5]The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. [6]Wide & Deep Learning for Recommender Systems.
[7]Improving forecasting accuracy of time series data using a new ARIMA-ANN hybrid method and empirical mode decomposition.
[8]Neural machine translation by jointly learning to align and translate.