第 48 卷第 5 期
2016 年 10 月

南 京 航 空 航 天 大 学 学 报
Journal of Nanjing University of Aeronautics & Astronautics

Vol. 48 No. 5
Oct. 2016

DOI:10.16356/j.1005-2615.2016.05.010

一种基于语义分析的大数据视频标注方法

崔 桐 徐 欣

(中国电子科技集团公司第二十八研究所,南京,210007)

摘要:提出一种基于 Spark 计算框架的海量视频语义标注方法。将存储在 Hadoop 分布式文件系统(Hadoop distributed file system,HDFS)上的海量视频部署到若干计算节点上,依据分形特征实现镜头快速分割。提取样本关键帧的颜色、纹理和分形特征向量,进行元学习策略训练,进而形成视觉词典。根据视觉词典对检测视频内容进行分析,产生一系列能表征视频内容的视觉单词。根据重要程度,通过马尔科夫链按重要程度对视频的视觉单词进行排序,并将排列结果作为该视频的标注。最后,从检测正确率、平均运行时间和扩展效能方面与传统分布式计算模型进行了对比。

关键词:大数据;视频标注;语义分析;元学习

中图分类号:TN915 **文献标志码**:A **文章编号**:1005-2615(2016)05-0677-06

Big Data Video Annotation Based on Semantic Analysis

Cui Tong, *XuXin*

(The 28th Research Institute, China Electronics Technology Group Corporation, Nanjing, 210007, China)

Abstract: A semantic annotation method is presented for massive videos based on Spark computing model. These massive videos are stored on Hadoop distributed file system (HDFS) and distributed to several nodes. These shot segmentationsin videos are realized by the fractal dimension method, and then the key frames of video shots are extracted based on color features, texture features and fractal features. The features in shots are trained using meta-learning strategy, and changed to video words and collected into the visual video dictionary. So video content is predicted and expressed by several video words according to the video dictionary. Then the video words are arranged according to importance sequence by Markov chains and the important words are described as video content prediction. Compared with the traditional distributed computing model, the Spark computing method illustrates the superiority from the correct rate, the average running time and the expansion efficiency.

Key words: big data; video annotation; semantic analysis; meta-learning

近年来,随着多媒体应用及社交网络的风靡,视频数据呈现指数级别的爆炸式增长,如何高效检索视频内容并标注,已成为大数据、机器视觉及多媒体应用领域的研究热点。早期,视频标注大多采用人工方法,费时费力,尤其面对海量视频,处理更是十分困难。对此,国内外诸多学者将图像处理、机器学习以及自然语言处理等技术结合起来,采用有监督学习方法[1-4]进行视频自动标注研究。例如,文献[5]尝试贝叶斯分类器对医疗教育视频中的语义概念进行分类;文献[6]提出经典主动学习

基金项目:国家自然科学基金(61402426)资助项目。

收稿日期:2015-10-01;**修订日期**:2016-06-30

通信作者:崔桐,男,高级工程师,E-mail:cuitong_seu@sina.cn。

引用格式:崔桐,徐欣. 一种基于语义分析的大数据视频标注方法[J]. 南京航空航天大学学报,2016,48(5):677-682.
Cui Tong, Xu Xin. Big data video annotation based on semantic analysis[J]. Journal of Nanjing University of Aeronautics & Astronautics, 2016,48(5):677-682.

方法，按照最小化期望分类误差准则选择样本，进而进行视频内容的预测；文献[7]使用自适应支持向量机进行跨域视频概念检测，使得分类器具备不同测试域间的自适应能力。

上述方法在数据量不大、实时性要求不高的情况下，能取得不错的效果。但对于海量视频，有限资源无法支持大规模运算，这些方法的应用受到制约。近年来，新兴大数据技术为海量视频标注提供了一条有效途径，不但解决了大容量视频数据的存储问题，而且，分布式计算也有利于视频语义的快速分析。其代表性工具 Spark[8-10] 是 UC Berkeley AMP lab 开源的并行计算框架，在机器学习处理方面具有独特优势，特别适合解决多次迭代的视频分析问题。因此，本文提出基于 Spark 的视频标注方法，利用其强计算能力，通过颜色、纹理、分形三重特征表征一类实体，进而采用元学习策略进行训练及预测。相对于传统分布式方法，该方法在标注效能方面有较大提升。

1 分布式镜头分割计算

视频通常是由一系列镜头构成，镜头又由连续拍摄且时间上连续的若干视频帧组成。为标注视频，需将视频分割成多组镜头的集合，提取出能够代表镜头内容信息的关键帧。因此，准确、快速地检测出镜头边界对视频语义表达具有重要影响。

本文在 Spark 分布式计算框架下，采用分形差分盒法[11]进行视频分割。首先，转换视频数据格式，将 Hadoop 分布式文件系统(Hadoop distributed file system，HDFS)上二进制的视频数据通过输出流转换为 Spark 可读取的数据，即通过函数 SparkContext 将视频数据读取为 String 类型 RDD。利用并行函数 Parallelize 把视频切分为以帧为单位的帧 RDD，并调用帧处理程序，将帧 RDD 数据并行分配到若干计算节点，通过 SparkContext 实现各个计算节点间镜头分割参数共享，从而使整个视频的帧数据实现并行处理。

在每个计算节点上，对视频帧采用差分盒法计算分形维度 D_i，定义第 i 帧的分形维度为 D_i，则第 i 帧与第 $i+1$ 帧的分形维度差可表示为 $fd_i=|D_{i+1}-D_i|$。在同一个镜头内，分形维度的帧差变化应在很小范围内，镜头边界帧差应远大于帧前镜头的帧差平均值和帧后镜头的帧差平均值。对于切变镜头，迭代求解出最大帧差 $fd_{\max}$，帧前镜头的帧差平均值 fd_{b_avg}，帧后镜头的帧差平均值 fd_{a_avg}；如果 $fd_{\max}>2*fd_{b\_avg}$ 且 $fd_{\max}>2*fd_{a\_avg}$，则判定该帧是切变镜头边界帧。对于渐变镜头，当渐变未被标记时，若 $fd_{\max}>2*fd_{b\_avg}$ 且 $fd_{\max}<2*fd_{a\_avg}$，则判定为渐变镜头边界的开始帧；如果渐变开始帧已被标记，若 $fd_{\max}>2*fd_{a\_avg}$，则判定为渐变镜头边界的结束帧，依此可将视频按照时间序列切分为若干镜头。

当视频处理完成后，视频每一帧均转化为[帧序号　分形维度]弹性分布式数据集(Resilient distributed dataset，RDD)数据，返回 Spark 主节点的结果是一组时间序列临界帧(即关键帧)的帧号和其帧 RDD 数据。将该 RDD 数据存储为关键帧文件，该文件包含视频关键帧的属性信息，具体过程如图 1 所示。

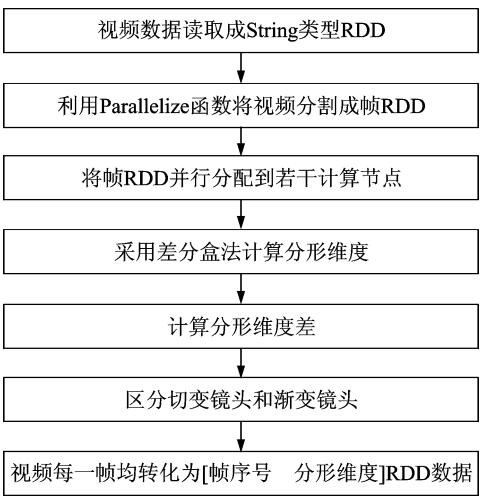

图 1 分布式镜头分割计算流程图

Fig. 1 Flow chart of distributed shot segmentation

2 视觉词典与视觉单词

视频标注是对视频数据进行处理、分析，并在理解内容的基础上，进行标记注释的过程。本文在 Spark 集群上提取视频对象的颜色、纹理及分形特征，通过元学习策略训练，形成视觉词典；并依据视觉词典对关键帧进行预测，产生能表征该镜头的视觉单词。

2.1 帧特征提取

选取实体对象的大量各异图片，提取其底层特征，包括 9 维颜色特征、8 维纹理特征、1 维分形维度特征，组成 18 维特征向量。其中，9 维颜色特征包括 3 个颜色分量，每个分量上 3 个低阶矩；8 维纹理特征考虑 Gabor 滤波器方向参数，可取 0°，22.5°，67.5°，90°，112.5°，135°，157.5°等 8 个方向参数。由于分形维度能更好的表示图像特征，赋予 32% 的较大权重，其他各维均匀赋予 4% 的权重，组成属性均衡的特征向量，表征该对象的视觉特征。将原始样本图片通过 pipe 函数分配到若干计

算节点进行特征提取,并将特征向量 RDD 数据存储到样本特征文件中,该文件包含实体对象的特征信息。

2.2　构建视觉词典

元学习[12]是在学习结果的基础上,再进行学习或多次学习得到最终结果的方法。在元学习中,使用不同的特征描述集合,能够有效减少基分类器输出结果的相关性,并使基分类器的错误相互独立[13-14]。同时,利用元学习实现算法自由参数的自动调整,即通过学习过程中获得的经验对这些参数进行修正和优化,从而提高学习算法的性能。

利用 2.1 节中提取的样本特征,通过 SparkContext 函数将样本特征文件读取为 RDD 数据,并分配到若干计算节点。以支持向量机(Support vector machine,SVM)、条件随机域(Conditional random field ,CRF)和最大熵(Maximum entropy,ME)作为基分类器,基于元学习方法对样本特征向量进行训练,实体样本 x_i(i 为样本序号),其表征特征向量为 $\mathbf{Vec}(x_i)$,正确分类标识为 $\boldsymbol{I}(x_i)$。通过上述 3 种基分类器训练,分别获得基分类模型 M_{svm},M_{rcf} 和 M_{me}。将 3 种算法的预测结果 $P(x_i)_{\text{svm}}$,$P(x_i)_{\text{rcf}}$,$P(x_i)_{\text{me}}$和 $\mathbf{Vec}(x_i)$,$\boldsymbol{I}(x_i)$作为元分类器样本 T,以 SVM 为元分类器进行二次训练,可得元分类模型M_{meta},如图 2 所示。不同于基分类器是将原始样本集作为输入,元分类器的样本 T 增加了基分类器的分类结果。样本集合 T 中存在 3 类样本:(1)所有基分类器皆分类正确;(2)所有基分类器皆分类错误;(3)基分类器结果存在矛盾。元分类器并不是从各个基学习器中挑选最佳学习器,而是对基学习器的结果进行“再学习”,对基学习器错误的分类进行纠正,而对正确的分类加以巩固,因此分类结果优于所有基分类器。训练得到元分类模型M_{meta}的表征 XML 文件,内含一个多维向量,该向量表示该类特征向量的视觉单词,将每个视觉单词与文字语义关联,使得每一个视觉单词(XML 文件)都与其文字符号相对应,录入视觉单词库。依此类推,对多种实体样本进行训练,进而累积形成视觉词典。

2.3　视觉单词的预测

依据视觉词典,采用元学习策略对实体对象进行预测。假设待测关键帧对象qx_i,表征特征向量为 $\mathbf{Vec}(qx_i)$。通过 2.2 节所述 3 种分类器预测,并将预测结果$Q(x_i)_{\text{svm}}$,$Q(x_i)_{\text{rcf}}$,$Q(x_i)_{\text{me}}$和 $q\mathbf{Vec}(x_i)$输入分类模型N_{meta},对照视觉词典中单词遍历预测是否包含该单词内容,如图 3 所示。一个关键帧中可能包含一个或多个视觉单词,程序返回 Spark 主机的结果是帧号、视觉单词、对应文字符号组成的 RDD 数据,将该数据存储为单词预测文件。

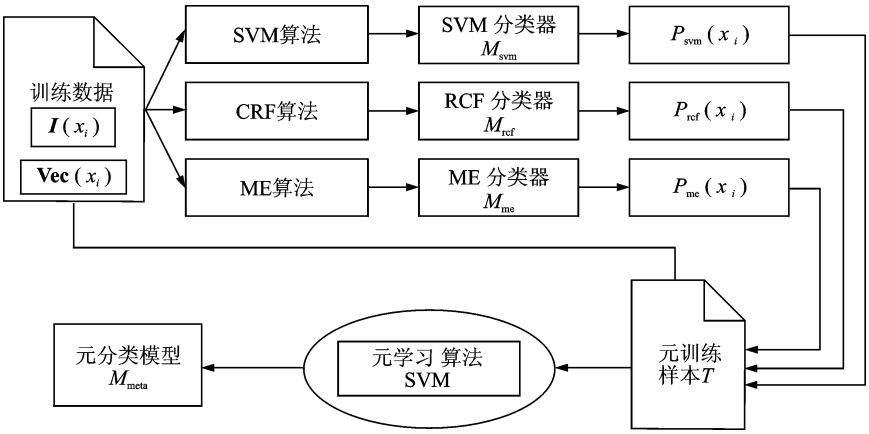

图 2　基于 Spark 的元学习训练过程

Fig. 2　Meta learning training process based on Spark

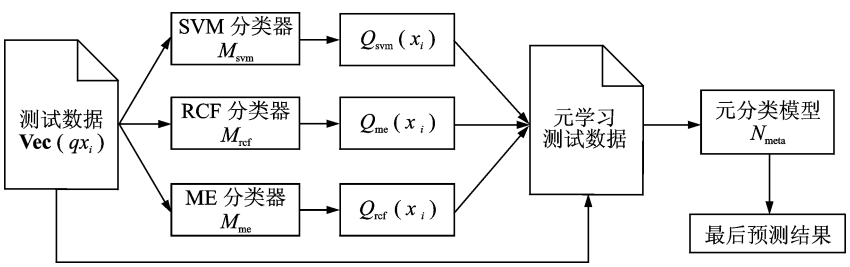

图 3　基于 Spark 的元学习预测过程

Fig. 3　Meta learning prediction process based on Spark

2.4 视频标注组成

将视频中各关键帧对应的视觉单词进行汇总，根据重要程度，通过马尔可夫模型[15]，对关键帧内容进行评估，实现基于视觉单词的线性表达，从而形成视频标注。

通过读取单词预测文件，利用 RDD 中键值统计函数 ReduceByKey，对每一关键帧所属视觉单词进行统计。假设视觉词典一共包含 N 个视觉单词，则每个关键帧包含视觉单词的集合为 $W=\{W^1,W^2,\cdots,W^i,\cdots,W^N\}$，$1\leq i\leq N$（当包含该单词时，$W^i$ 为 1，否则为 0），则 k 个关键帧所包含的视觉单词可组成转移矩阵 $\boldsymbol{M}$，可表示为

$$\boldsymbol{M}=\begin{bmatrix} W_1^1 & W_2^1 & \cdots & W_k^1 \\ W_1^2 & W_2^2 & \cdots & W_k^2 \\ \vdots & \vdots & \ddots & \vdots \\ W_1^N & W_2^N & \cdots & W_k^N \end{bmatrix} \tag{1}$$

初始化时，假定 k 个关键帧中，每个关键帧对视频内容重要性系数均为 $1/k$，则 k 个关键帧组成系数均为 $1/k$ 的 k 维列向量 $\boldsymbol{V}_1$。当所有关键帧都包含两个及以上视觉单词时，用 $\boldsymbol{V}_1$ 右乘转移矩阵 $\boldsymbol{M}$，可得

$$\boldsymbol{V}_2=\boldsymbol{M}\boldsymbol{V}_1=\begin{bmatrix} W_1^1 & W_2^1 & \cdots & W_k^1 \\ W_1^2 & W_2^2 & \cdots & W_k^2 \\ \vdots & \vdots & \ddots & \vdots \\ W_1^N & W_2^N & \cdots & W_k^N \end{bmatrix}\begin{bmatrix} \frac{1}{k} \\ \frac{1}{k} \\ \vdots \\ \frac{1}{k} \end{bmatrix} \tag{2}$$

当关键帧只包含一个视觉单词时，为保证迭代收敛，设定该视觉单词是视频内容的概率是 α，则

$$\boldsymbol{V}_2=\alpha\boldsymbol{M}\boldsymbol{V}_1+(1-\alpha)\boldsymbol{V}_1=$$

$$\alpha\begin{bmatrix} W_1^1 & W_2^1 & \cdots & W_k^1 \\ W_1^2 & W_2^2 & \cdots & W_k^2 \\ \vdots & \vdots & \ddots & \vdots \\ W_1^N & W_2^N & \cdots & W_k^N \end{bmatrix}\begin{bmatrix} \frac{1}{k} \\ \frac{1}{k} \\ \vdots \\ \frac{1}{k} \end{bmatrix}+(1-\alpha)\begin{bmatrix} \frac{1}{k} \\ \frac{1}{k} \\ \vdots \\ \frac{1}{k} \end{bmatrix} \tag{3}$$

经过 n 次收敛的迭代，递归矩阵 $\boldsymbol{M}\boldsymbol{V}_n$ 最终将收敛为一个 k 维向量，该向量中数值越大的元素表示对应关键帧对视频内容表征越重要，其所含视觉单词越能代表视频的内容。

3 实验分析

实验采用大小为 1 TB 的视频集为实验对象，主要来源于监控视频、世界军事等视频资料，包括 5 000 余个视频训练镜头和 100 个测试镜头，每个镜头时长 5～10 min 不等。鉴于视频相邻帧之间的关联性与相似性，实验中采用跳帧法，以提高运行效率。选取视频帧底层特征，包括 9 维的颜色特征、8 维纹理特征和 1 维分形维度特征，组成 18 维特征向量，赋予分形特征 32%的权重，其他特征均匀赋予 4%的权重，从而构建该帧的视觉特征。

测试过程主要从检测正确率、运行效率和平台扩展性 3 个方面进行性能评估。检测正确率、运行效率多与算法本身设计、迭代方式以及并行化实现原理有关；平台扩展性则更多地体现并行化效能。实验集群环境共 22 台 PC 机器，配置如表 1 所示。

各机器上安装 Spark 版本 1.0.0，Hadoop 版本采用 1.0.1，JDK 环境为 OpenJDK 1.7.0 64 位版本。主要针对内存占用量配置，在 Spark 框架中每个结点的计算内存为 20 G，提供计算数据以及 RDD 线性依赖关系存储所占用的空间。

表 1 实验集群硬件配置

Tab. 1 Experimental cluster hardware configuration

编号节点名称	CPU	数量	内存容量/GB	硬盘容量/GB
Master	酷睿 i7-3820 8 核	1	64	1 024
Slave	至强 E31230 8 核	21	32	500

3.1 视频分段解码

在视频解码过程中，运用 Spark 框架，通过自定义的 InputFormat 接口读取视频文件，实现对于视频文件分块读取及同步处理。在解码过程中，InputFormat 为解码程序提供容量固定的数据片段（实验中采用 4 MB/分片），分发到不同的节点进行处理。对 HDFS 中的数据分片 FileSplit 进行排序，使所有视频文件开头的第一个数据块最先读取，并将包含视频解码必需的视频头信息作为后续数据的固有前缀，以提高分片解码效率。如图 4 所示，假设一个视频文件有 A，B，C 等多个数据块并分布在 HDFS 上，其中数据块 C 是包含了视频文件的头信息，通过数据块排序和分隔技术，将数据块分割为 D_1，D_2，…，D_n 等数据片，把视频头信息

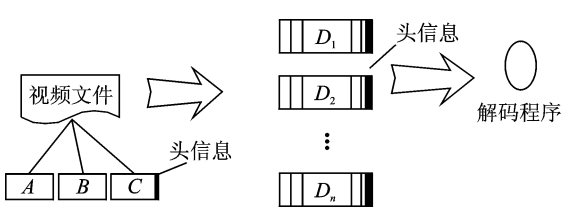

图 4 视频分段解码示意图

Fig. 4 Schematic diagram of video segmentation decoding

读取并存于内存中，对每个数据分片添加视频头信息，整合后的数据分片可交由解码程序进行解码。

3.2　正确率检测

首先通过 Spark 计算框架，结合分形差分盒法对人脸、飞机、坦克和行人等 4 类视频进行镜头分割实验。镜头分割的效果度量标准有查全率和查准率，其中：查全率＝正确检测数/(正确检测数＋漏检数)，查准率＝正确检测数/(正确检测数＋误检数)，实验结果如表 2 所示，查全率达到 97%以上，查准率达到 90%以上，可以满足视频实体目标提取的要求。

表 2　镜头分割实验参数

Tab. 2　Experimental parameters of shot segmentation

视频	帧数	镜头数	检测数	正确数	漏检数	误检数	查全率/%	查准率/%
人脸	370	11	12	11	0	1	100.00	91.67
飞机	1 579	39	53	38	1	3	97.43	92.68
坦克	1 922	67	81	65	2	5	97.01	92.85
行人	1 074	37	42	36	1	2	97.30	94.74

实验对每个镜头提取一副关键帧，进行颜色、边界和分形特征进行提取，形成 18 维特征向量，对视频中的人脸、飞机、坦克和行人等 4 类实体对象进行训练并测试，将视觉单词在视频画面中进行标定，如图 5 所示。

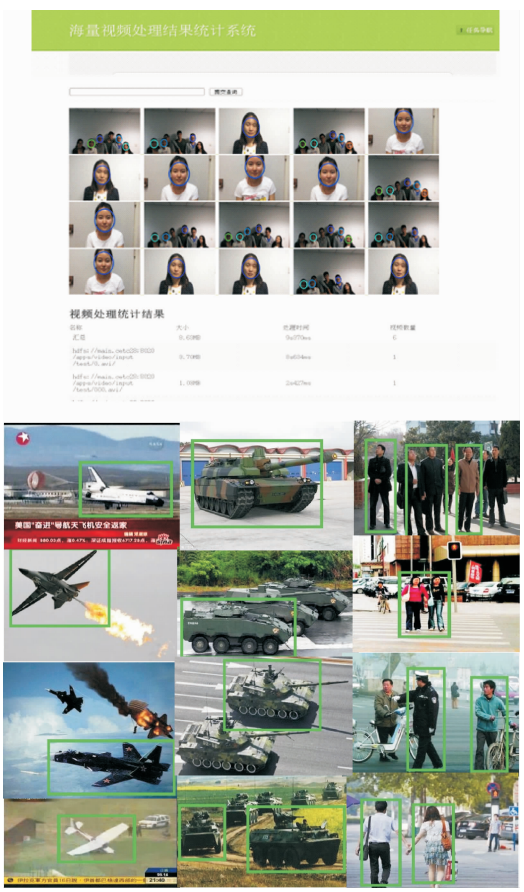

图 5　视觉单词的标定

Fig. 5　Calibration of visual words

分类过程选取 SVM，CRF 和 ME 为元分类器，并采用高斯核 SVM 为决策分类器。如图 6 所示，人脸、飞机、坦克和行人的检测正确率相对于文献[3]提出的自适应 SVM 方法分别提升了 5%，7%和 8%，行人的预测精度提升了 1%。

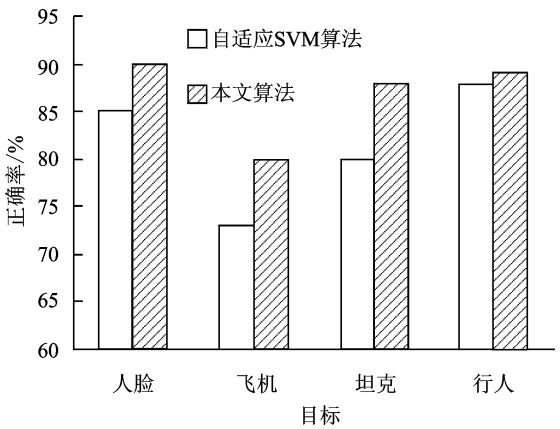

图 6　检测正确率对比

Fig. 6　Comparison of detection accuracy

3.3　运行时间

迭代运行的元策略算法需要多次遍历数据集，每轮的迭代运行开销很大程度上决定了算法运行的总体时间。但由于迭代时间不宜统计，且与物理机性能关系密切。因此，在对比 Spark 框架与 MapReduce 框架实现算法时，对比了在相同数据量运算条件下，运行总时间的差别，从中能够很明显地查看出 Spark 执行时间上远小于 MapReduce，如表 3 所示。

表 3　运算时间

Tab. 3　Operation time　　s

数据量/GB	100	500	800	1 000
MapReduce	220.1	586.2	1 052.6	1 584.2
Spark	3.5	85.6	142.6	231.8

3.4　扩展性

随着数据量的增加，MapReduce 框架和 Spark 框架在运行效率上均表现出平稳性。为了对比两者的线性扩展能力，利用 16～22 个集群计算节点，测试 1 TB 视频数据，比较 MapReduce 框架

与 Spark 框架在不同节点数量下的加速比，如表 4 所示。

表 4 平均运行时间

Tab. 4 Average running time s

节点	16	18	20	22
MapReduce	1 586.4	1 396.2	1 238.9	1 125.3
Spark	265.8	220.4	165.1	131.1

从表 4 可以看出，当前该方法具有很好的水平扩展能力。随着计算节点数的增加，投入任务更多的计算资源，计算任务的运行时间可以呈现很明显的下降趋势。加速比的结果反映了并行化实现的效率提升情况，Spark 并行化实现方法能极大地提高程序运行效率，性能得到显著的改进。

4 结束语

本文提出了一种基于 Spark 的海量视频语义标注方法，采用 Spark 计算框架，实现对体量庞大的非结构化视频数据进行分析，将海量复杂多源的视频数据转化为机器可识别的、具有明确语义的信息，进行视频标注。实验证明，相对于传统分布式框架计算计算模型，这种方法具有迭代速度快，扩展性能强等优点。

参考文献：

[1] Chorianopoulos K, Giannakos M N, Chrisochoides N, et al. Open service for video learning analytics[C]//Proceedings of the 14th International Conference on Advanced Learning Technologies. Athens, Greece: [s. n.], 2014: 28-30.

[2] Lu Shiyang, Wang Zhiyong, Mei Tao, et al. A bag-of-importance model with locality-constrained coding based feature learning for video summarization[J]. IEEE Transactions on Multimedia, 2014, 16(6): 1497-1509.

[3] Md Noor N, Hamizan N I, Ab Rahim R. The framework for learning using video based on cognitive load theory among visual learners[C]//2013 IEEE 5th Conference on Engineering Education. Kuala Lumpur, Malaysia: [s. n.], 2013: 15-20.

[4] Hung I C, Wei C W, Chen N S. Designing dynamic scaffolding strategy for improving video-based learning in a gesture and speech-based learning configuration[C]//Proceedings of the 13th International Conference on Advanced Learning Technologies. Beijing, China: [s. n.], 2013: 194-196.

[5] Fan Jianping, Luo Hangzai, Li Xiaodong. Semantic video classification by integrating flexible mixture model withadaptive EM algorithm[C]// Proceedings of the 5th ACM SIGMM International Workshop on Multimedia Information Retrievals. New York, USA: [s. n.], 2003: 9-16.

[6] Ding Youdong, Zhang Jianfei, Li Jun,, et al. Bag-of-feature model for video semantic annotation[C]// Proceedings of 6th International Conference on Imageand Graphics. Hefei, China: [s. n.], 2011: 696-701.

[7] Huang T M. Kernel based algorithms for mining huge data sets: Supervised, semi-supervised, and unsupervised learning[M]. Berlin, Germany: Springer-Verlag, 2010: 297-301.

[8] Lin Xiuqin, Wang Peng, Wu Bin. Log analysis in cloud computing environment with Hadoop and Spark[C]//Proceedings of the 5th IEEE International Conference on Broadband Network & Multimedia Technology. Guilin, China: Institute of Electrical and Electronics Engineers, 2013: 273-276.

[9] Gu Lei, Li Huan. Memory or time: Performance evaluation for iterative operation on Hadoop and Spark[C]//High Performance Computing and Communications & 2013 IEEE International Conference on Embedded and Ubiquitous Computing. Zhangjiajie, China: Research Gate, 2013: 721-727.

[10] 姚远，梁志毅. 基于差分盒维数的空间目标图像分割算法[J]. 计算机科学, 2012, 39 (11A): 359-362.
Yao Yuan, Liang Zhiyi. Image segmentation for space target based on different box counting[J]. Computer Science, 2012, 39(11A): 359-362.

[11] Liu Xuan, Wang Xiaoguang, Matwin S, et al. Meta-learning for large scale machine learning with map reduce[C]//Proceedings of the 2013 IEEE International Conferenceon Big Data. CA, USA: [s. n.], 2013: 105-110.

[12] Chekina L, RokachL, Shapira B. Meta-learning for selecting a multi-labelclassification algorithm[C]//Proceedings of the 11th International Conference on Data Mining. Las Vegas, USA: [s. n.], 2011: 220-227.

[13] Shilbayeh S, Vadera S. Feature selection in meta learning framework[C]//Proceedings of the 2014 Science and Information Conference. London, England: [s. n.], 2014: 269-275.

[14] Pesce M, Munaretto D, Zorzi M. A Markov decision model for source video rate allocation and scheduling policies in mobile networks[C]//Proceedings of the 13th Annual Mediterranean Ad Hoc Networking Workshop. Piran, Slovenia: Institute of Electrical and Electronics Engineers, 2014: 119-125.