Falco: A quick and flexible single-cell RNA-seq processing framework on the cloud

Andrian Yang[1,2], Michael Troup[1], and Joshua WK Ho[1,2]

[1]Victor Chang Cardiac Research Institute, Sydney, NSW, Australia
[2]St. Vincent's Clinical School, University of New South Wales, Sydney, NSW, Australia

July 15, 2016

Abstract

Summary: Single-cell RNA-seq (scRNA-seq) is increasingly used in a range of biomedical studies. Nonetheless, current RNA-seq analysis tools are not specifically designed to efficiently process scRNA-seq data due to their limited scalability. Here we introduce Falco, a cloud-based framework for parallelised processing of large-scale transcriptomic data. The pipeline utilises state-of-the-art big data technology of Apache Hadoop and Apache Spark to perform massively parallel alignment, quality control, and feature quantification of single-cell transcriptomic data in Amazon Web Service (AWS) cloud-computing environment. We have evaluated the performance of Falco using two public scRNA-seq datasets and demonstrated Falco's scalability. The result shows Falco performs at least 2.6x faster against a highly optimized single node analysis and a reduction in runtime with increasing number of computing nodes. Falco also allows user to the utilise low-cost spot instances of AWS, providing a 65% reduction in cost of analysis.

Availability: Falco is available via an open source license in https://github.com/VCCRI/Falco/

Contact: j.ho@victorchang.edu.au

Supplementary information: Supplementary data are available at *BioRXiv* online.

1 Introduction

Major advancements in single-cell technology have resulted in the increasing interest in single-cell level studies, particularly in the field of transcriptomics [8]. Single-cell RNA sequencing (scRNA-seq) offers the promise of understanding transcriptional heterogeneity of individual cells, allowing for a clearer understanding of biological process [11, 7, 3, 9].

Each scRNA-seq experiment typically generates profiles of hundreds of cells, which is a magnitude larger than the typical amount of data generated by standard bulk RNA-seq experiments. Current RNA-seq processing pipelines are not specifically designed to handle such a large number of profiles. To fully realise the potential of scRNA-seq, we need a scalable and efficient computational solution.

The premise of our solution is that state-of-the-art cloud computing technology, which is known for its scalability, elasticity and pay-as-you-go payment model, can allow for a highly scalable and efficient scRNA-seq analysis.

There are a number of existing cloud-based next-generation sequencing bioinformatics tools. They are based on the Hadoop framework, an open source implementation of MapReduce [4], or the Spark framework [13], such as Halvade [5], SparkSeq [12] and SparkBWA [1]. Halvade is designed mainly to perform variant calling of genomic data from raw FASTQ files, though it also offers support for transcriptomic analysis. SparkSeq offers interactive sequencing analysis of BAM files while SparkBWA only supports alignment of FASTQ files. These tools have limitations in the context of scRNA-seq analysis. Of the three tools, only SparkSeq allows for multi-sample analysis, although SparkSeq itself is also limited as it does not perform alignment, which is the main bottleneck in sequence analysis.

To address the limitations of existing tools for the analysis of scRNA-seq datasets, we introduce Falco, a cloud-based pipeline for simultaneous processing of large-scale transcriptomic data geared for multi-sample analysis.

2 Framework

Falco is a framework built for the Amazon Web Services (AWS) cloud-computing platform. It utilises Amazon Elastic MapReduce (EMR), a managed big data processing software framework which allows for easy deployment and access to Apache Hadoop and Apache Spark cluster.

Falco itself consists of a splitting step, an optional pre-processing step and the main analysis step (Supplementary Figure 3). The first step, the

splitting step, is a MapReduce job which splits FASTQ input files stored in the Amazon S3 storage service into multiple smaller FASTQ files. In the case of paired-end reads, the two reads are combined into a single record to ensure that paired-end reads are processed together. The splitting process is performed in order to increase the level of parallelism in analysis and normalise the performance of tools as each chunk will have the same maximum uncompressed size of 256 MB.

The next step in the pipeline is an optional step for performing pre-processing of reads, such as adapter trimming and filtering reads based on quality. The pre-processing step is another MapReduce job which performs pre-processing of the split FASTQ files using any pre-processing tools chosen by the user. The user is asked to supply a shell script with commands to run their selected pre-processing tools, that is then called by the MapReduce job.

The final step of the pipeline is the main analysis step. It performs alignment and quantification of reads using the Spark framework. In the current implementation, each split FASTQ file is aligned using STAR [6] and quantified using featureCounts [10]. The returned gene counts per split are then reduced (*i.e.*, merged) to obtain the total read counts per gene in each sample. The gene count matrix is produced and stored into Amazon S3 storage. Aside from the gene counts, the analysis step also returns selected mapping and quantification reports generated by STAR and featureCounts as well as optional RNA-seq alignment metrics from Picard tools [2].

As part of the pipeline, a script is provided to simplify the creation of the EMR cluster and configure the required software and references on the cluster. Similarly, each of the steps also has a corresponding submission script which will upload the files required for the step and submit the step to the EMR cluster for execution.

3 Evaluation

To evaluate the performance of Falco, a comparison was performed between the runtime of the pipeline and the performance of a single large computing resource to which an organisation with limited computing resources may be expected to have. A number of realistic scenarios for analysis in a single computing node were devised - from the naive single processing approach to a highly parallelised approach. Furthermore, to demonstrate the scalability of Falco, EMR clusters with increasing numbers of core nodes (from 10 to 40) were used to show the effect of adding more computational resources on

Table 1: Analysis completion time - standalone vs Falco

System	Nodes	Mouse - embryonic stem cell (hours)	Human - brain (hours)
Standalone	1 (1 process)	93.7	154.7
	1 (5 processes)	29.3	33.8
	1 (12 processes)	21.1	16.4
	1 (16 processes)	18.5	13.6
Falco	10	7.0	2.7
	20	4.1	1.6
	30	3.3	1.4
	40	2.8	1.1

Standalone number of processes indicates the number of FASTQ file pairs that are processed in parallel. Timing for Falco includes initialisation and configuration time.

the runtime of Falco.

In both the single resource and the scalability test comparison, the AWS EC2 instance type used for computation (core node for EMR) is r3.8xlarge (32 cores, 244GB of RAM and two 320GB SSDs). For Falco's EMR cluster, a single r3.4xlarge (16 cores, 122GB RAM) was used as the master node for scheduling jobs and managing the cluster. The EMR cluster uses Amazon EMR release 4.6, which contains Apache Hadoop 2.7.2 and Apache Spark 1.6.1, and takes 16 minutes for initialisation and configuration in all cluster configurations used.

Two recently published scRNA-seq datasets were used for evaluation. The first dataset (SRA accession: ERP005988), is a mouse embryonic stem cell (mESC) single cell data containing 869 samples of 200bp paired-end reads, totalling to 1.28×10^{12} sequenced bases, stored in 1.02 Tb of gzipped FASTQ files [9]. The second dataset (SRA accession: SRP057196), is a smaller human brain single cell data containing 466 samples of 100 bp paired-end reads, totalling to 2.95×10^{11} sequenced bases and 213.66 Gb of gzipped FASTQ files [3].

Comparing the performance of a single node, with different parallelisation approaches, against Falco shows that Falco outperforms the single node by a minimum of 2.6x (10 node vs 16 process) and a maximum of 33.4x (40 node vs 1 process) for the mouse dataset and a minimum of 5.1x (10 node vs 16 process) and a maximum of 145.4x (40 node vs 1 process) for the human dataset (Table 1). The disparity in the speed-up between the two datasets is due to different pre-processing tools being employed, with the

Table 2: Falco cost analysis - on-demand vs spot instances

Dataset	Cluster size	Time (hours)	On-demand cost (USD)	Spot cost (USD)	% Savings
Mouse -	10 node	8	247.20	85.67	65.34
embryonic	20 node	5	301.00	99.09	67.08
stem cell	30 node	4	258.00	115.71	55.15
	40 node	3	356.40	114.11	67.98
Human -	10 node	3	92.70	32.13	65.34
brain	20 node	2	120.40	39.64	67.08
	30 node	2	179.00	57.86	67.68
	40 node	2	237.60	76.08	67.98

Time rounded up to whole hour including cluster startup. Includes the cost of the master node. Price used for r3.8xlarge instance is 2.660 (on-demand price) and 0.64 (average spot price for June 2016) - USD/hour.

human data set utilising multiple pre-processing tools as specified in the original publication [3].

For the scalability comparison, it can be seen that the runtime of the pipeline decreases with increasing cluster size (Table 1), though the trend is gradual rather than linear. Analysis of the runtime for each step in the framework shows a similar gradual decrease in runtime for pre-processing and analysis steps (Supplementary Figure 2). For the splitting step, a different trend is seen where there is little to no decrease in runtime for cluster size $\geq$ 20 nodes. The lack of speed up for splitting is due to the number of executors exceeding the number of files to be split and the limitation of time taken to split large files as the distribution of file size in both test datasets is uneven (Supplementary Figure 1).

To save cost, EMR allows for the usage of reduced price *spot* computing resources. The spot prices fluctuate depending on the availability of the unused computing resource and the spot instance is obtained by supplying a bid for the resource. As shown in (Table 2), the use of spot instances for analysis provide a substantial saving of around 65% compared to using on-demand instances. The trade-off with using spot instances is that the computing resource could be terminated should the market price for that resource exceed the user's bid price.

4 Summary

Falco is a cloud-based framework designed for multi-sample analysis of transcriptomic data in an efficient and scalable manner.

Funding

This work was supported in part by funds from the New South Wales Ministry of Health, a National Health and Medical Research Council/National Heart Foundation Career Development Fellowship (1105271), a Ramaciotti Establishment Grant (ES2014/010), an Australian Postgraduate Award, and Amazon Web Services (AWS) Credits for Research.

References

[1] J. M. Abuín, J. C. Pichel, T. F. Pena, and J. Amigo. SparkBWA: Speeding Up the Alignment of High-Throughput DNA Sequencing Data. *PLOS ONE*, 11(5):e0155461, may 2016.

[2] Broad Institute. Picard tools - by broad institute, 2016.

[3] S. Darmanis, S. A. Sloan, Y. Zhang, M. Enge, C. Caneda, L. M. Shuer, M. G. Hayden Gephart, B. A. Barres, and S. R. Quake. A survey of human brain transcriptome diversity at the single cell level. *Proceedings of the National Academy of Sciences*, 112(23):7285–7290, jun 2015.

[4] J. Dean and S. Ghemawat. MapReduce: Simplified Data Processing on Large Clusters. *Communications of the ACM*, 51(1):107, jan 2008.

[5] D. Decap, J. Reumers, C. Herzeel, P. Costanza, and J. Fostier. Halvade: scalable sequence analysis with MapReduce. *Bioinformatics*, 31(15):2482–2488, aug 2015.

[6] A. Dobin, C. a. Davis, F. Schlesinger, J. Drenkow, C. Zaleski, S. Jha, P. Batut, M. Chaisson, and T. R. Gingeras. STAR: ultrafast universal RNA-seq aligner. *Bioinformatics (Oxford, England)*, 29(1):15–21, jan 2013.

[7] D. Grün, A. Lyubimova, L. Kester, K. Wiebrands, O. Basak, N. Sasaki, H. Clevers, and A. van Oudenaarden. Single-cell messenger RNA sequencing reveals rare intestinal cell types. *Nature*, 525(7568):251–255, aug 2015.

[8] A. A. Kolodziejczyk, J. K. Kim, V. Svensson, J. C. Marioni, and S. A. Teichmann. The Technology and Biology of Single-Cell RNA Sequencing. *Molecular Cell*, 58(4):610–620, may 2015.

[9] A. A. Kolodziejczyk, J. K. Kim, J. C. Tsang, T. Ilicic, J. Henriksson, K. N. Natarajan, A. C. Tuck, X. Gao, M. Bühler, P. Liu, J. C. Marioni, and S. A. Teichmann. Single Cell RNA-Sequencing of Pluripotent States Unlocks Modular Transcriptional Variation. *Cell Stem Cell*, 17(4):471–485, oct 2015.

[10] Y. Liao, G. K. Smyth, and W. Shi. FeatureCounts: An efficient general purpose program for assigning sequence reads to genomic features. *Bioinformatics*, 30(7):923–930, 2014.

[11] A. P. Patel, I. Tirosh, J. J. Trombetta, A. K. Shalek, S. M. Gillespie, H. Wakimoto, D. P. Cahill, B. V. Nahed, W. T. Curry, R. L. Martuza, D. N. Louis, O. Rozenblatt-Rosen, M. L. Suva, A. Regev, and B. E. Bernstein. Single-cell RNA-seq highlights intratumoral heterogeneity in primary glioblastoma. *Science*, 344(6190):1396–1401, jun 2014.

[12] M. S. Wiewiorka, A. Messina, A. Pacholewska, S. Maffioletti, P. Gawrysiak, and M. J. Okoniewski. SparkSeq: fast, scalable and cloud-ready tool for the interactive genomic data analysis with nucleotide precision. *Bioinformatics*, 30(18):2652–2653, sep 2014.

[13] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica. Spark : Cluster Computing with Working Sets. In *Proceedings of the 2Nd USENIX Conference on Hot Topics in Cloud Computing*, page 10. USENIX Association, 2010.