Doctoral Consortium: Paralelización y adaptación de algoritmos de mantenimiento y detección de fallos a plataformas de cómputo en la nube

Mario Juez-Gil
Área de Lenguajes y Sistemas Informáticos, Departamento de Ingeniería Civil
Universidad de Burgos
Burgos, España
mariojg@ubu.es

***Resumen*—El incremento de los volúmenes de información con los que se trabaja en minería de datos, abre nuevas vías de investigación en ámbitos como la paralelización de algoritmos. Surgen también, nuevas posibilidades de aplicación en entornos industriales, como, por ejemplo, en tareas como mantenimiento y detección de fallos en entornos industriales. Este proyecto de tesis tiene como finalidad explorar y desarrollar técnicas de minería de datos paralelizables para su ejecución en arquitecturas paralelas como GPUs o plataformas de cómputo en la nube, para su posterior aplicación industrial.**

***Index Terms*—Parallelism, cloud computing, CUDA, Map-Reduce, data mining, multi-label, ensembles, fault diagnosis**

I. DATOS

I-A. *Datos de contacto*

- Nombre y apellidos: Mario Juez Gil
- Dirección: Barriada Juan XXIII, nº 1, 8º, 3ª. 09007, Burgos
- Teléfono: +34 652213401
- Correo electrónico: mariojg@ubu.es
- LinkedIn: /mjuez

I-B. *Propuesta de título para la tesis*

Paralelización y adaptación de algoritmos de mantenimiento y detección de fallos a plataformas de cómputo en la nube.

I-C. *Directores*

- César Ignacio García Osorio (cgosorio@ubu.es)
- Carlos López Nozal (clopezno@ubu.es)

I-D. *Departamento*

Departamento de Ingeniería Civil, Área de Lenguajes y Sistemas Informáticos, Universidad de Burgos.

I-E. *Fecha de Inicio*

7 de Marzo de 2018.

II. RESUMEN

La minería de datos se centra en el estudio y tratamiento de grandes cantidades de datos para extraer conclusiones e información relevante y comprensible de conjuntos de datos para su uso posterior. Dos de las técnicas más comunes en la minería de datos son:

- Clasificación: Se asume que los datos pertenecen a distintas clases caracterizadas por los valores de sus atributos. El objetivo es construir clasificadores/modelos que asignen la etiqueta de la clase correcta a nuevas clases no etiquetadas.
- Regresión: En este caso, el atributo a determinar no es cualitativo o discreto (clase), sino numérico o continuo.

La aplicación de técnicas de minería de datos puede ser de utilidad en la mejora del mantenimiento y disponibilidad de maquinaria industrial. Utilizando los datos consistentes en las medidas tomadas por los sensores instalados en las máquinas, y la información de los históricos de las acciones de mantenimiento y reparación de averías, se puede asistir en la predicción de las averías y fallos, o incluso automatizar totalmente la predicción, permitiendo adelantar las acciones de preparación de recursos materiales y humanos para acometer el mantenimiento.

En este tipo de máquinas podrían ocurrir varias averías o fallos de forma simultánea (el fallo de un componente podría desencadenar fallos en otros componentes relacionados), por tanto, será necesario obtener varias predicciones (tantas como posibles componentes afectados) para un mismo conjunto de datos de entrada. En minería de datos, estos supuestos se conocen como problemas de salida múltiple o problemas de aprendizaje multietiqueta [1]–[3].

Las técnicas de clasificación multietiqueta y de regresión multivalor, aunque tienen un ambito de aplicación amplio, su proceso de adopción está siendo lento. En el campo de las aplicaciones industriales apenas hay referencias. Entre las que conocemos, en [4] se utiliza la predicción multietiqueta para detección de fallos simultáneos en plantas químicas, y en [5]

en el campo de las perturbaciones en la calidad del suministro eléctrico. Sin embargo, en ningún caso nos constan trabajos de aplicación en la línea de los que propone el presente proyecto, lo que refuerza su carácter innovador.

Una forma de abordar los problemas de salida múltiple, es a través de la utilización de algoritmos de ensembles, los cuales permiten obtener predicciones mediante la combinación de varios modelos, homogéneos o heterogéneos [6]. Existe un consenso generalizado en que estos métodos son la mejor técnica para tratar los problemas más difíciles, ya que consiguen un mejor rendimiento frente al uso de un modelo único [7]–[9], además, el grupo de investigación en el que me encuentro desarrollando mi actividad investigadora, tiene amplia experiencia en el diseño de este tipo de métodos.

El tiempo de respuesta en tareas de predicción debe ser aceptable, no tendría sentido obtener el resultado de una predicción una vez que ya haya ocurrido el suceso a predecir. Cuando se trabaja con grandes volúmenes de datos, los tiempos de respuesta pueden ser elevados, por lo que se debe poner especial atención en tratar de reducirlos en la medida de lo posible.

La evaluación de posibilidades para mejorar el rendimiento de algoritmos de minería de datos a través de la paralelización de los mismos, es el objetivo principal de esta tesis. Este tipo de técnicas resulta interesante porque los algoritmos de ensembles son inherentemente paralelizables, cada clasificador base se puede entrenar de forma independiente en la mayoría de los métodos de construcción de ensembles (a excepción de *boosting* y sus derivados), para después combinar sus resultados. Pero es que, además, en el contexto de aprendizaje multietiqueta surgen nuevas oportunidades de paralelización y particionado de los procesos de cómputo. Por tanto, la paralelización de los algoritmos puede hacerse a varios niveles distintos, lo cual creemos que puede tener un impacto muy positivo en la reducción de los tiempos de respuesta. La paralelización de algoritmos también será especialmente relevante para poder ofrecer servicios de *Machine Learning as a Service* mediante la ejecución de los algoritmos desarrollados por el grupo de investigación en plataformas de computación paralela en la nube.

II-A. *Objetivos*

La tesis tiene dos objetivos generales:

- Objetivo 1: Mejora de rendimiento de algoritmos para su explotación en tareas de mantenimiento y detección de fallos en entornos industriales.
- Objetivo 2: Facilitar el acceso a los algoritmos a través de servicios Web desplegados en plataformas de cómputo en la nube.

Como se ha descrito anteriormente, para mejorar la eficiencia y escalabilidad de los algoritmos, se utilizarán técnicas de paralelización. El primer objetivo general tiene los siguientes objetivos específicos:

- Subobjetivo 1.1: Adaptación de algoritmos para explotación de paralelismo mediante su ejecución en GPUs.
- Subobjetivo 1.2: Adaptación de algoritmos para explotación de paralelismo mediante su ejecución aplicando el modelo de cómputo *Map-Reduce*.

El segundo objetivo general, para poder ofrecer *Machine Learning as a Service*, plantea los siguientes objetivos específicos:

- Subobjetivo 2.1: Desarrollo de servicios Web (por ejemplo, APIs REST) que expongan interfaces públicas para permitir el uso de los algoritmos desarrollados.
- Subobjetivo 2.2: Integración de los algoritmos paralelizados con los servicios Web desarrollados mediante su ejecución en plataformas de cómputo en la nube.

III. Metodología y Plan de trabajo

III-A. *Metodología*

La paralelización de algoritmos consiste en desarrollar un *software*, por tanto, se seguirán los principios metodológicos de la Ingeniería del Software.

Debido a que este trabajo de investigación es una parte de un proyecto en el que participa el director, se seguirá su misma metodología para la extracción de conocimiento de los datos aplicando técnicas de aprendizaje automático, denominada *KDD process* (*Knowledge Discovery in Databases*) [10].

También se utilizarán metodologías de evaluación de rendimiento de los métodos desarrollados, haciendo uso de medidas para paralelización [11], y para clasificación multietiqueta y predicción multivalor, como, por ejemplo, las propuestas en [12], [13].

III-B. *Plan de trabajo*

Este proyecto de tesis busca explorar principalmente dos tipos de esquemas de paralelización: esquemas basados en *Map-Reduce* en computación en la nube; y esquemas basados en paralelismo que ofrecen las GPUs. Adicionalmente también se estudiarán posibilidades híbridas que combinen ambos tipos de esquemas. Las fases previstas (o hitos) de la tesis serán las siguientes:

1. Estudio de bibliotecas de paralelización en GPUs.
2. Estudio de biblioteca Spark MLlib de paralelización *Map-Reduce*.
3. Desarrollo de prototipos e implantación en industria de algoritmos paralelos para GPU y Spark MLlib.
4. Evaluación y comparativa de resultados obtenidos con ambos esquemas de paralelización. Estudio de posibilidades de combinación de esquemas.
5. Adaptación de algoritmos desarrollados por el grupo de investigación a plataformas de cómputo paralelo en la nube.

A continuación, se describe con más detalle la finalidad de cada una de las fases:

III-B1. Estudio de bibliotecas de paralelización en GPUs: Las GPUs son un tipo de *hardware* inicialmente diseñado para el renderizado de gráficos e impulsado por la industria de los videojuegos. Sin embargo, actualmente es posible hacer uso de lenguajes de alto nivel para programar algoritmos que puedan

ser ejecutados en arquitecturas GPU gracias a bibliotecas como CUDA de NVIDIA [14]. Existen artículos [15], [16] donde se aboga por la paralelización en GPUs como la mejor solución para conseguir algoritmos de minería de datos y aprendizaje automático más eficientes. En esta fase del trabajo de investigación se estudiarán bibliotecas como CUDA [17] o OpenCL [18], así como las posibilidades existentes para su uso con algoritmos de ensembles y aprendizaje multietiqueta.

III-B2. Estudio de biblioteca de Spark MLlib de paralelización Map-Reduce: Uno de los paradigmas de computación paralela más populares es *Map-Reduce* [19], implementaciones como Spark (https://spark.apache.org/) permiten incrementar el rendimiento de algoritmos de minería de datos y aprendizaje automático. Una búsqueda en Google del término *«spark machine learning»* arroja más de cuatro millones de resultados, y muestra que MLlib (https://spark.apache.org/mllib/) es la biblioteca donde parece obligado proporcionar implementaciones de los algoritmos desarrollados por el grupo, si se quiere aumentar el impacto que éstos puedan tener en la comunidad científica. En esta fase del trabajo de investigación se estudiará la biblioteca MLlib y las posibilidades existentes para su uso con algoritmos de ensembles y aprendizaje multietiqueta.

III-B3. Desarrollo de prototipos e implantación en industria de algoritmos paralelos para GPU y Spark MLlib: Tras adquirir los conocimientos necesarios sobre el funcionamiento de los dos esquemas principales de paralelización de algoritmos y sus bibliotecas, se llevarán a la práctica mediante el desarrollo de prototipos de algoritmos de minería de datos paralelos que puedan ser implantados en entornos industriales. La finalidad de esta fase es doble: por un lado se buscará demostrar la validez de nuestros algoritmos paralelos en tareas de mantenimiento y detección de fallos en entornos industriales, a través de publicaciones en revistas científicas; mientras que por otra parte, también se pretende compartir los conocimientos adquiridos con el resto de miembros del grupo de investigación.

Los prototipos también serán útiles para evaluar el rendimiento de ambos paradigmas en fases futuras.

III-B4. Evaluación y comparativa de resultados obtenidos con ambos esquemas de paralelización. Estudio de posibilidades de combinación de esquemas: Los prototipos desarrollados se evaluarán teniendo en cuenta distintas medidas de rendimiento. Tras la evaluación se hará una comparativa entre los resultados obtenidos empleando paralelización en GPU, y aquellos obtenidos empleando Spark MLlib, con el fin de escoger el esquema con mejor rendimiento para su posterior aplicación en tareas de mantenimiento y detección de fallos en entornos industriales. Para medir el rendimiento de los algoritmos desarrollados se tendrá en cuenta tanto su naturaleza paralela, como multietiqueta. Por ello se utilizarán métricas como rapidez (*speedup*) o eficiencia [11] para el primer caso, y métricas como *F measure*, precisión, o *Hamming loss* [12], [13] para el segundo.

En esta fase también se estudiarán las posibilidades existentes para combinar ambos esquemas de paralelización, como, por ejemplo, la ejecución de Spark en GPUs propuesta por [20].

III-B5. Adaptación de algoritmos desarrollados por el grupo de investigación a plataformas de cómputo paralelo en la nube: Recientemente han surgido diversas plataformas de cómputo en la nube, entre las que destacan Google Cloud Platform, Amazon AWS, o Microsoft Azure. Este tipo de plataformas permite configurar agrupaciones (*clusters*) de cómputo con un número de nodos personalizado, lo cual es de gran utilidad para la ejecución de algoritmos paralelos de minería de datos. Surgen así, posibilidades para ofrecer como servicio (*Machine Learning as a Service*) los algoritmos y tecnologías desarrollados por el grupo, de manera que puedan ser utilizados fácilmente por empresas que necesiten de estos servicios. En esta fase del trabajo de investigación se adaptarán los algoritmos de minería de datos del grupo para su ejecución en plataformas de cómputo en la nube a través de servicios públicos, como por ejemplo APIs REST [21].

Cuadro I
HITOS DE LA TESIS

Hito	Objetivo	Descripción
1	**1**	**Estudio bibliotecas de paralelización**
	1.1	Estudio bibliotecas GPU
	1.2	Estudio bibliotecas Spark
2	**1**	**Desarrollo de prototipos e implantación industrial**
	1.1	Implementación de prototipo GPU
	1.1	Publicación proponiendo la implantación del protitipo GPU en entornos industriales
	1.2	Implementación de prototipo Spark MLlib
	1.2	Publicación proponiendo la implantación del protitipo Spark MLlib en entornos industriales
3	**1**	**Evaluación y comparativa de resultados**
	1.1	Evaluación prototipo GPU
	1.2	Evaluación prototipo Spark MLlib
	1.1, 1.2	Comparativa resultados GPU vs MLlib
	1.1, 1.2	Estudio posibilidad híbrida GPU + Spark
4	**2**	**Adaptación de algoritmos del grupo a la nube**
	2.1	Desarrollo de servicios web públicos
	2.2	Integración de algoritmos con servicios web

III-C. Hitos

Los hitos de la tesis y su relación con los objetivos se describen en la Tabla I.

La tabla está dividida en cuatro secciones, cada una de las cuales se corresponde con un hito general. La segunda columna indica sobre qué objetivo/subobjetivo de los descritos en II-A se está trabajando. La última columna es una breve descripción del hito.

IV. RELEVANCIA

Trabajar con cantidades de datos cada vez mayores es una tendencia imparable en la actualidad. Surgen por tanto, nuevos retos científicos, como, por ejemplo, la necesidad de una evolución de las técnicas existentes de minería de datos para que sean escalables y capaces de manejar grandes volúmenes de información. Dicho objetivo se puede conseguir mediante la paralelización de estos algoritmos, utilizando modelos ya

existentes como *Map-Reduce* o desarrollando nuevas soluciones paralelas en GPUs.

Nuestra investigación ofrece además, posibilidades de aplicación en la industria, concretamente en tareas como la detección temprana de fallos en maquinaria industrial, donde optimizar el tiempo de respuesta de los algoritmos, o aplicar técnicas multietiqueta (por ejemplo, para la predicción de múltiples combinaciones de fallos), puede tener un impacto muy positivo.

Referencias

[1] E. Gibaja and S. Ventura, "Multi-label learning: a review of the state of the art and ongoing research," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 4, no. 6, pp. 411–444, nov 2014. [Online]. Available: http://doi.wiley.com/10.1002/widm.1139

[2] ——, "A Tutorial on Multilabel Learning," *ACM Computing Surveys*, vol. 47, no. 3, pp. 1–38, apr 2015. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2737799.2716262

[3] G. Madjarov, D. Kocev, D. Gjorgjevikj, and S. Džeroski, "An extensive experimental comparison of methods for multi-label learning," *Pattern Recognition*, vol. 45, no. 9, pp. 3084–3104, 2012.

[4] I. Yélamos, M. Graells, L. Puigjaner, and G. Escudero, "Simultaneous fault diagnosis in chemical plants using a multilabel approach," *AIChE Journal*, vol. 53, no. 11, pp. 2871–2884, 2007.

[5] Z. Liu, Y. Cui, and W. Li, "A classification method for complex power quality disturbances using EEMD and rank wavelet SVM," *IEEE Transactions on Smart Grid*, vol. 6, no. 4, pp. 1678–1685, 2015.

[6] L. I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms*. New York, NY, USA: Wiley-Interscience, 2004.

[7] E. Bauer and R. Kohavi, "An empirical comparison of voting classification algorithms: Bagging, boosting, and variants," *Machine learning*, vol. 36, no. 1, pp. 105–139, 1999.

[8] O. Maimon and L. Rokach, *Data Mining and Knowledge Discovery Handbook*, 2nd ed. Springer Publishing Company, Incorporated, 2010.

[9] G. I. Webb, "Multiboosting: A technique for combining boosting and wagging," *Machine learning*, vol. 40, no. 2, pp. 159–196, 2000.

[10] U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, Eds., *Advances in Knowledge Discovery and Data Mining*. Menlo Park, CA, USA: American Association for Artificial Intelligence, 1996.

[11] J. JáJá, *An Introduction to Parallel Algorithms*. Redwood City, CA, USA: Addison Wesley Longman Publishing Co., Inc., 1992.

[12] H. Borchani, G. Varando, C. Bielza, and P. Larrañaga, "A survey on multi-output regression," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 5, no. 5, pp. 216–233, sep 2015. [Online]. Available: http://doi.wiley.com/10.1002/widm.1157

[13] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Information Processing & Management*, vol. 45, no. 4, pp. 427–437, jul 2009. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0306457309000259

[14] L. Jian, C. Wang, Y. Liu, S. Liang, W. Yi, and Y. Shi, "Parallel data mining techniques on Graphics Processing Unit with Compute Unified Device Architecture (CUDA)," *The Journal of Supercomputing*, vol. 64, no. 3, pp. 942–967, jun 2013. [Online]. Available: http://link.springer.com/10.1007/s11227-011-0672-7

[15] J. Nickolls and W. J. Dally, "The GPU Computing Era," *IEEE Micro*, vol. 30, no. 2, pp. 56–69, mar 2010. [Online]. Available: http://ieeexplore.ieee.org/document/5446251/

[16] D. Singh and C. K. Reddy, "A survey on platforms for big data analytics," *Journal of Big Data*, vol. 2, no. 1, p. 8, dec 2015. [Online]. Available: http://www.journalofbigdata.com/content/2/1/8

[17] Nvidia, "Nvidia cuda c programming guide," 01 2010.

[18] J. E. Stone, D. Gohara, and G. Shi, "OpenCL: A Parallel Programming Standard for Heterogeneous Computing Systems," *Computing in Science & Engineering*, vol. 12, no. 3, pp. 66–73, may 2010. [Online]. Available: http://ieeexplore.ieee.org/document/5457293/

[19] J. Dean and S. Ghemawat, "MapReduce," *Communications of the ACM*, vol. 51, no. 1, p. 107, jan 2008. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1327452.1327492

[20] P. Li, Y. Luo, N. Zhang, and Y. Cao, "HeteroSpark: A heterogeneous CPU/GPU Spark platform for machine learning algorithms," in *2015 IEEE International Conference on Networking, Architecture and Storage (NAS)*, 2015, pp. 347–348.

[21] M. Masse, *REST API Design Rulebook: Designing Consistent RESTful Web Service Interfaces*. O'Reilly Media, Inc., 2011.