Adaptación del algoritmo Teoría de Respuesta al Ítem para la estimación del conocimiento latente sobre Datos Educacionales Masivos

Adaptation of the algorithm Item Response Theory for the latent knowledge estimation about Massive Educational Data

Ing. Orlando Grabiel Toledano López

Dr C. Rafael Arturo Trujillo Rasúa

MSc. Angel Alberto Vazquez Sánchez

Universidad de las Ciencias Informáticas. La Habana, Cuba

Universidad de las Ciencias Informáticas. La Habana, Cuba

Universidad de las Ciencias Informáticas. La Habana, Cuba

Resumen

La constante interacción tecnológica de las personas, los dispositivos móviles y estaciones de trabajo conectadas a la red genera un gran cúmulo datos, de los cuales se puede extraer conocimiento. El sector educacional no está ajeno a este fenómeno y se puede observar cómo el uso creciente de las Tecnologías de la Información y las Comunicaciones influyen en sus procesos sustanciales. El área de la Minería de Datos Educativos permite aplicar métodos computarizados para la realización de tareas predictivas que posibilitan detectar patrones sobre datos educativos y clasificar grupos de individuos a partir de la medición de su conocimiento latente. Entre los métodos predictivos, existe Teoría de Respuesta al Ítem, el cual es un modelo matemático de medición que permite conocer, a partir de respuestas observadas de individuos sobre ítems evaluativos, la habilidad o conocimiento latente. Este no es viable aplicarlo en su estado actual sobre cantidades masivas de datos, debido a que su costo computacional depende de la cantidad de estudiantes y la cantidad de ítems. Para dar respuesta a este problema se realiza una adaptación de dicho modelo mediante el uso de técnicas de procesamiento en paralelo usando la herramienta *Apache Spark*. Para ello se analizan sus elementos matemáticos fundamentales y técnicas de estimación de parámetros de un ítem para identificar las partes paralelizables de este.

Finalmente, como aporte práctico de la solución se obtiene un algoritmo de estimación disponible a través de una API REST de servicios, con un mejor rendimiento que el algoritmo original.

Palabras clave: Algoritmo, Minería de Datos Educativos, Teoría de Respuesta al Ítem.

Abstract

The constant technological interaction of people, mobile devices and work stations connected to the network generates a large accumulation of data, from which knowledge can be extracted. The educational sector is not exempt to this phenomenon and it can be seen how the growing use of Information and Communication Technologies influence its substantial processes. The area of Educational Data Mining allows applying computerized methods to perform predictive tasks that make it possible to detect patterns on educational data and classify groups of individuals based on the measurement of their latent knowledge. Among the predictive methods, there is Item Response Theory, which is a mathematical measurement model that allows knowing, from the observed responses of individuals on evaluative items, the skill or latent knowledge. This is not viable to apply in its current state on massive amounts of data, because its computational cost depends on the number of students and the number of items. In order to respond to this problem, an adaptation of said model is made through the use of parallel processing techniques using the Apache Spark tool. To do this, we analyze its fundamental mathematical elements and techniques for estimating the parameters of an item to identify the parallel parts of it. Finally, as a practical contribution of the solution, an estimation algorithm available through a service REST API is obtained with a better performance than the original algorithm.

Keywords: *Algorithm, Educational Data Mining, Item Response Theory*

Introducción

En la actualidad, la constante interacción tecnológica de las personas, los dispositivos móviles y estaciones de trabajo conectadas a la red genera un gran cúmulo de huellas digitales o datos, de las cuales se puede extraer conocimiento útil y comprensible. Para lograr adquirir y analizar tanta información surge el área de conocimiento conocida como *Big Data*, la cual se caracteriza por conjuntos de datos cuyo tamaño va más allá de la capacidad de captura, almacenado, gestión y análisis de las herramientas de base de datos. Big Data puede ser utilizado para la mejora de procesos permitiendo su simplificación y control, propiciando la reducción de costes y esfuerzos (Camargo-Vega, Camargo-Ortega, & Joyanes-Aguilar, 2015). Además, para identificar patrones de comportamiento complejos que permitan detectar ataques informáticos en tiempo real. Enfocado al soporte a la toma de decisiones, puede ser también una potente herramienta para descubrir información que antes podría estar oculta, a través de la aplicación algoritmos automáticos de minería de datos. Todas estas ventajas se pueden agrupar en una principal de las que derivan el resto: "obtener más información/conocimiento" a partir del análisis profundo de los datos (Manyika, Chui, & Brown, 2011).

La Educación, como sector impulsor y enriquecedor de la sociedad, no ha quedado exenta del impacto del desarrollo tecnológico, quedando expuesta, además, a las potencialidades del *Big Data* y su uso enfocado a fortalecer el proceso de enseñanza aprendizaje. Este fenómeno se puede observar en dicho sector debido al uso creciente de las TIC en sus procesos vitales, donde la interacción con herramientas educativas en líneas, sistemas adaptativos, Cursos Online Masivos y Abiertos (MOOCs, siglas en inglés),

entre otras plataformas, genera un gran cúmulo de datos educacionales que pueden ser aprovechados para encontrar conocimiento oculto y mejorar de forma sustancial y efectiva el proceso educativo docente (Santos, 2015).

En el marco del contexto del análisis de datos masivos surge la Minería de Datos Educacionales (MDE), la cual se ocupa del desarrollo, investigación y aplicación de métodos computarizados para detectar patrones en grandes colecciones de datos que provienen del contexto educacional, que de otra manera podría ser difícil o imposible de analizar debido al enorme volumen en los cuales estos existen. Estos datos provienen de varias fuentes, incluidos los datos de los entornos presenciales tradicionales, el uso de software educativo, los cursos en línea y las actividades evaluativas (Romero C. , Ventura, Pechenizkiy, Baker, & S.J.d., 2011).

Con el uso de la MDE se pueden aplicar diferentes métodos para descubrir conocimiento sobre grandes fuentes de datos, tales como: Métodos predictivos, descubrimiento de estructuras y modelos, minería de relaciones y destilación de datos para juicio humano (Galindo & García, 2010). Los datos educacionales masivos comprenden datos de identidad, datos de interacción de usuarios del sistema, datos de todo el sistema y datos inferidos por los estudiantes. A partir de todos ellos se puede extraer conocimiento de un gran valor (Romero & Ventura, 2012).

Entre los métodos predictivos existentes, resulta de gran utilidad para el sector educacional la estimación de conocimiento latente, el cual permite determinar en qué medida un estudiante domina un conocimiento o habilidad determinada en un momento dado (J. A. Larusson, 2014). Para realizar una inferencia del conocimiento que posee un estudiante, entiéndase por conocimiento lo que un estudiante conoce (o domina) que puede ser significativo para la situación de aprendizaje actual (Habilidades, Hechos, Conceptos, Principios y Esquemas) existen diferentes algoritmos, tales como: *Performance Factors Analysis* (PFA) (Goldin, 2015), *Bayesian Knowledge Tracing* (BKT) (Sande, 2013), *Item Response Theory* (IRT) (Yue & K, 2017). Estos proveen modelos donde a partir de datos dicotómicos[1] obtenidos por diferentes test aplicados a los estudiantes permiten estimar conocimiento latente. Con ellos, de forma general, se puede medir qué conoce un estudiante en un momento específico, sus componentes de conocimiento y estimar su probabilidad de éxito frente a una pregunta o ítem en un examen (BAKER & al., 2010).

De los algoritmos anteriormente planteados, particularmente IRT se puede aplicar en la elaboración de instrumentos evaluativos dinámicos y adaptables a las habilidades individuales de los estudiantes. Se debe tener en cuenta, que este parte de la rama de la psicología para el diseño de test mentales que permiten la medición del conocimiento humano (Pérez Gil, 2011). Desde sus inicios, ha sido una buena alternativa al modelo Teoría Clásica de Test (TCT) que era utilizado con el mismo propósito. Este supone una mejora con respecto a TCT, ya que ofrece métodos que permiten la construcción de test más adecuados, eficientes y permite evaluar en una misma escala a los ítems evaluativos y los examinados (BAKER F. B., 2001). Lo más importante de IRT es que asegura que las mediciones de conocimiento latente realizadas a las personas sean invariantes con respecto a los ítems aplicados en la medición (Güler, Uyanık, & Teker, 2014).

Aplicar IRT, en su estado actual, para el procesamiento de cantidades masivas de datos no es viable debido a que su costo computacional depende del tamaño de los datos de entrada, dígase cantidad de estudiantes a evaluar e ítems evaluativos de los instrumentos aplicados en la medición. Esto ocurre debi-

1 **Datos dicotómicos:** Registro de vectores de respuestas con valores binarios (ceros y uno). Se usa el valor cero si ha contestado incorrectamente y uno si ha contestado correctamente.

do a que las plataforma educativas que constituyen fuentes de datos para la estimación de conocimiento interactúan de forma diaria miles de estudiantes sobre un número considerable de cursos educativos que poseen ítems agrupados en instrumentos de evaluación, además estos pueden crecer periódicamente. Un ejemplo de esto lo constituye la plataforma *Coursera* y según datos recogidos hasta el 2015 esta alcanzó los 18 millones de usuarios de todo el mundo (Coursera, 2016).Cada uno de estos usuarios es evaluado constantemente por los instrumentos de los propios cursos, por lo que medir conocimiento para todos ellos constituye un verdadero desafío.

Por lo anterior se plantea el siguiente problema de la investigación: ¿Cómo incrementar el rendimiento computacional del algoritmo Teoría de Respuesta al Ítem para aplicarlo en la estimación de conocimiento latente sobre Datos Educacionales Masivos?

A partir de la problemática planteada el objetivo de este trabajo será desarrollar una adaptación del algoritmo Teoría de Respuesta al Ítem, usando técnicas de procesamiento en paralelo y distribuidas, para aumentar el rendimiento computacional en la estimación de conocimiento latente sobre Datos Educacionales Masivos.

Materiales y métodos o Metodología computacional

Para un mayor entendimiento del algoritmo IRT se procede a definir el modelo matemático que permite la estimación de conocimiento latente para un examinado conociendo las respuestas observadas sobre los ítems evaluativos. Las respuestas observadas representan un vector $U = (u_1, u_2, \dots, u_i), i = 1, 2, 3, \dots, N$ $U = (u_1, u_2, \dots, u_i). i = 1, 2, 3, \dots, N$, siendo NN la cantidad de ítems. Cada valor $u_i u_i$ es un valor de respuesta dicotómico, donde $u_i\, u_i$ toma valor 1 si la respuesta es correcta y 0 si la respuesta es falsa.

Modelo IRT para la estimación de conocimiento latente y modelos logísticos de estimación

El modelo IRT surge para solventar algunas limitaciones que tenía TCT, sus inicios datan alrededor de 1960 y fue planteado por *George Rasch* y se basa en dos supuestos fundamentales (Pérez Gil, 2011):

- El atributo que se desea medir puede representarse en una única dimensión en la que se situarían conjuntamente las personas y los ítems.
- El nivel de una persona en el atributo y la dificultad del ítem determina la probabilidad de que la respuesta sea correcta.

Bajo estos supuestos *Rasch* planteó una ecuación logística que permite estimar la probabilidad de que un sujeto responda correctamente un ítem, denominado también modelo de un parámetro, pues solo tiene en cuenta la dificultad del ítem, partiendo siempre del nivel de conocimiento o rasgo latente del individuo.

El modelo logístico de un parámetro se plantea de la siguiente forma (L. Thorpe & Favia, 2012):

$$\boldsymbol{P(X_i = 1|\theta) = (1 + e^{-(\theta-\beta_i)})^{-1} P(X_i = 1|\theta) = (1 + e^{-(\theta-\beta_i)})^{-1}} \quad \textit{[1]}$$

Donde $\beta_i\beta_i$ indica la dificultad del ítem ii,$\theta\theta$ indica el nivel de habilidad del individuo.

Esto no quedó limitado a un parámetro, pues en el contexto de la evolución de los test intervienen otras variables que no se pueden despreciar en la medición, tales como: la probabilidad de adivinar un ítem o contestar correctamente al azar y el índice de discriminación. Estos nuevos modelos fueron propuestos por *Birnbaum* y *Fred Lord*; los cuales logran un mejor tratamiento matemático y facilitan el desarrollo de nuevos métodos de estimación (Pérez Gil, 2011).

El modelo de dos parámetros se plantea de la siguiente manera (L. Thorpe & Favia, 2012):

$$\boldsymbol{P(X_i = 1|\theta) = (1 + e^{-a_i(\theta-\beta_i)})^{-1}P(X_i = 1|\theta) = (1 + e^{-a_i(\theta-\beta_i)})^{-1}}$$

[2]

Donde se mantiene la dificultad del ítem $\beta_i\beta_i$ y la habilidad del examinado $\theta\theta$. Además se incluye el parámetro discriminador a_ia_i, el cual representa la magnitud de cambio en la probabilidad de acertar el ítem ii, conforme varía el nivel de habilidad del individuo (Pérez Gil, 2011).

El modelo de tres parámetros se plantea de la siguiente manera (L. Thorpe & Favia, 2012):

$$\boldsymbol{P(X_i = 1|\theta) = c + (1 - c)(1 + e^{-a_i(\theta-\beta_i)})^{-1}}$$

$$\boldsymbol{P(X_i = 1|\theta) = c + (1 - c)(1 + e^{-a_i(\theta-\beta_i)})^{-1}} \qquad \textit{[3]}$$

Donde se tiene además el coeficiente de azar o pseudoazar cc, indicando la probabilidad de contestar correctamente al azar el ítem ii.

Estimación de la habilidad usando Teoría de Respuesta al Ítem

La estimación del nivel de habilidad de un individuo a partir de los datos dicotómicos obtenidos por un test es una parte fundamental en IRT, pues conociendo los parámetros de cada ítem y calculando las probabilidades probabilidad de responder correctamente, por cualquiera de los modelos anteriores; se puede estimar el nivel de habilidad de un individuo, basándose en procedimientos de máxima verosimilitud. Esto se realiza mediante un proceso iterativo que comienza con un valor de habilidad inicial determinado *a priori* y un vector de respuestas dicotómico. (Natarajan, 2009).

La ecuación de estimación que se aplica en un algoritmo secuencial queda de la siguiente forma:

$$\boldsymbol{\theta_{s+1} = \theta_s + \sum_{i=1}^{n} a_i [u_i - P_i(\theta_s)] \Big/ \sum_{i=1}^{n} a_i{}^2 P_i(\theta_s) Q_i(\theta_s)}$$

$$\boldsymbol{\theta_{s+1} = \theta_s + \sum_{i=1}^{n} a_i [u_i - P_i(\theta_s)] \Big/ \sum_{i=1}^{n} a_i{}^2 P_i(\theta_s)} \qquad \textit{[4]}$$

Donde $\theta_s\theta_s$ es el nivel de habilidad de un individuo en la iteración ss, $P_i(\theta_s)P_i(\theta_s)$es la probabilidad de responder correctamente un ítem y $O_i(\theta_s)Q_i(\theta_s)$ es la probabilidad de no responder correctamente el ítem, se puede calcular mediante 1- $P_i(\theta_s)P_i(\theta_s)$ (BAKER F. B., 2001). Por otra parte se tiene en la ecuación la variable u_iu_i que

representa el i-ésimo valor dicotómico del vector de respuesta de del individuo sobre el ítem ii.

Para ejecutar el procedimiento de estimación aplicando la ecuación, inicialmente $\theta_s\theta_s$ se le otorga un valor arbitrario que puede ser 1. Luego se calcula la probabilidad de responder correctamente cada uno de los NN ítems aplicando la ecuación logística correspondiente al modelo (Ver ecuaciones 1,2 y 3) y se resuelven las dos expresiones de suma. En cada iteración se actualiza $\theta_{s+1}\theta_{s+1}$, para aplicarlo a las próximas iteraciones como estimación anterior. El procedimiento concluye cuando se alcance el error estándar deseado o no ocurran cambios significativos en los valores estimados (Natarajan, 2009).

Para el cálculo del error estándar como otro elemento a tener en cuenta en la condición de parada del procedimiento, se define la siguiente ecuación (Natarajan, 2009):

$$SE(\theta_s) = {1}\Big/{\sqrt{\sum_{i=1}^{n} a_i{}^2 P_i(\theta_s) Q_i(\theta_s)}} SE(\theta_s) = {1}\Big/{\sqrt{\sum_{i=1}^{n} a_i{}^2 P_i(\theta_s) Q_i(\theta_s)}}$$

[5]

El error estándar representa una medida de la variabilidad de $\theta\theta$alrededor del valor del parámetro desconocido del examinado, que es nivel de habilidad (Natarajan, 2009).

A partir de lo anterior se puede constatar que existen modelos matemáticos que permiten estimar el conocimiento latente de un individuo a partir de un vector de respuestas dicotómico que se obtiene sobre un grupo determinado de ítems, teniendo en cuenta sus parámetros. Se pueden aplicar ecuaciones logísticas para determinar la probabilidad de responder correctamente un determinado ítem dado un nivel de habilidad y observar la curva característica del mismo para su análisis. Con esto se pueden mejorar los instrumentos evaluativos, pudiéndose aplicar en un sistema evaluador exámenes que se adaptan dinámicamente (JF, J, M, D, D, & E., 2014).

Un problema fundamental para el desarrollo de la propuesta es el procesamiento de los vectores de respuestas dicotómicos que se obtienen con las respuestas observadas de los examinados sobre los ítems. Para a partir de estos, estimar los parámetros que poseen los ítems evaluativos según los modelos logísticos elegidos. Una vez estimados, se puede inferir el conocimiento latente que poseen los individuos evaluados y su probabilidad de responder correctamente frente a un ítem determinado. Para lograr dicho proceso existen métodos estadísticos adaptados a IRT, los cuales realizan una estimación a partir de las respuestas observadas (Melià, 2018).

Entre las técnicas para la estimación de parámetros en ítems dicotómicos para IRT se pueden mencionar las siguientes: Método de Máxima Verosimilitud Conjunta (JML, siglas en inglés), Estimación de Máxima Verosimilitud Condicional (CML, siglas en inglés), Máxima Verosimilitud Marginal (MML, siglas en inglés). De los anteriores el desarrollo de este trabajo se centra en el uso de MML de debido a que este método resulta adecuado cuando se desea estimar para los tres modelos logísticos, donde estos no tienen consistencia estructural y el número de casos aumenta. Los desarrolladores de este método (Bock y Lieberman, 1970) lograron integrarlo con el método *Expectation-Maximization* (EM) para que este funcionara correctamente para test evaluativos con un considerable número de ítems. Esta técnica es una de las más utilizadas para la estimación en IRT. Este proceso se realiza al integrar la función de verosimilitud con respecto a la distribución normal (la distribución de las habilidades de los examinados) y maximiza la verosimilitud con respecto a los parámetros de los ítems (Sehgal, G., & Garg, D. K., 2014)

Algoritmo secuencial de EM para el modelo IRT

EM es un algoritmo iterativo que busca estimadores de máxima verosimilitud o máxima verosimilitud a posteriori, para estimar parámetros en modelos estadísticos donde este depende de variables latentes no observadas. En cada iteración del algoritmo se ejecuta un paso de Expectación (E) y un paso de Maximización (M). Este proceso se repite hasta que se alcance un número determinado de iteraciones, se obtenga el error estándar deseado en la estimación o no ocurran ya cambios significativos en la estimación (Sehgal & Garg, 2014).

Para ello el algoritmo recibe como entrada una matriz YY de tamaño $N\ x\ IN\ x\ I$, donde NN representa el número de examinados y JJ el número de ítems. En la entrada $Y = (y_1, y_2, \dots, y_N)Y = (y_1, y_2, \dots, y_N)$, y_iy_i representa el vector de respuesta dicotómico del examinado $i\,i$ dado por $(y_{i1}, y_{i2}, \dots, y_{iJ})(y_{i1}, y_{i2}, \dots, y_{iJ})$ y $y_{ij}y_{ij}$ es un valor de respuesta dicotómico del estudiante $i\,i$ sobre el ítem jj (Harwell, 1988).

La salida del algoritmo es un vector de ítems con parámetros estimados $I = (\delta_1, \delta_2, \dots, \delta_j)I = (\delta_1, \delta_2, \dots, \delta_j)$ donde $\delta_j = \{t_1, t_2, \dots, t_T\}\delta_j = \{t_1, t_2, \dots, t_T\}$ representa el conjunto de TT parámetros del ítem jj.

La estimación comienza con una distribución de probabilidades condicionales de tamaño KK que representan los posibles valores de la variable latente $\theta_i\theta_i$ que es la habilidad del estudiante ii. Los valores de la distribución condicional están denotados por $q_k, k\epsilon[1;K]q_k, k\epsilon[1;K]$ (Harwell, 1988)}.

A continuación, se explica cómo se procede en cada paso (Sehgal & Garg, 2014):

- **Paso de Expectación (Paso E).** En este paso se crea una función para la expectación de verosimilitud de logaritmo natural, usando la estimación actual para los parámetros.
- **Paso de Maximización (Paso M).** Se calculan los parámetros deseados del modelo maximizando la función de verosimilitud de logaritmo natural calculada con el paso E.

Tabla 1. Algoritmo secuencial EM

Entrada	• Sea YY matriz de tamaño $N\ x\ JN\ x\ J$, donde NN representa el número de examinados y JJ el número de ítems. • Sea $I' = (\delta'_1, \delta'_2, \dots, \delta'_J)I' = (\delta'_1, \delta'_2, \dots, \delta'_J)$ vector de ítems con parámetros desconocidos, donde $\delta'_i = \{t_1, t_2, \dots, t_T\}\delta'_i = \{t_1, t_2, \dots, t_T\}$ representa el conjunto de TT parámetros de un ítem ii. • Sea μ máximo cambio • Sea ϕ cantidad máxima de iteraciones.
Salida	• Vector de ítems con parámetros estimados $I = (\delta_1, \delta_2, \dots, \delta_J)$ $I = (\delta_1, \delta_2, \dots, \delta_J)$

Pasos:

1. Inicializar $s = 0 s = 0$, $e = 1 + \ \mu e = 1 + \ \mu$
2. Inicializar $Q = (q_1, q_2, \ldots, q_K) Q = (q_1, q_2, \ldots, q_K)$, vector de distribución condicional.
3. Inicializar $\pi = (\pi_1, \pi_2, \ldots, \pi_K) \pi = (\pi_1, \pi_2, \ldots, \pi_K)$, vector de distribución de probabilidades asociada a la variable latente $k \epsilon [1; K] \ k \epsilon [1; K]$
4. Mientras $s < \phi s < \phi$ y $e > \ \mu e > \ \mu$
5. Paso E.

i. $$n_k^{(s)} = \sum_{i=1}^{N} \frac{\pi_k^{(s)} \prod_{j=1}^{J} P(q_k|\delta_j^{(s)})^{y_{ij}} [1-P(q_k|\delta_j^{(s)})^{y_{ij}}]^{1-y_{ij}}}{\sum_{k'=1}^{K} \pi_{k'}^{(s)} \prod_{j}^{J} P(q_{k'}|\delta_j^{(s)})^{y_{ij}} [1-P(q_{k'}|\delta_j^{(s)})^{y_{ij}}]^{1-y_{ij}}}$$

$$n_k^{(s)} = \sum_{i=1}^{N} \frac{\pi_k^{(s)} \prod_{j=1}^{J} P(q_k|\delta_j^{(s)})^{y_{ij}} [1-P(q_k|\delta_j^{(s)})^{y_{ij}}]^{1-y_{ij}}}{\sum_{k'=1}^{K} \pi_{k'}^{(s)} \prod_{j}^{J} P(q_{k'}|\delta_j^{(s)})^{y_{ij}} [1-P(q_{k'}|\delta_j^{(s)})^{y_{ij}}]^{1-y_{ij}}}$$

ii. $$r_{jk}^{(s)} = \sum_{i=1}^{N} \frac{y_{ij} \pi_k^{(s)} \prod_{j=1}^{J} P(q_k|\delta_j^{(s)})^{y_{ij}} [1-P(q_k|\delta_j^{(s)})^{y_{ij}}]^{1-y_{ij}}}{\sum_{k'=1}^{K} \pi_{k'}^{(s)} \prod_{j}^{J} P(q_{k'}|\delta_j^{(s)})^{y_{ij}} [1-P(q_{k'}|\delta_j^{(s)})^{y_{ij}}]^{1-y_{ij}}}$$

$$r_{jk}^{(s)} = \sum_{i=1}^{N} \frac{y_{ij} \pi_k^{(s)} \prod_{j=1}^{J} P(q_k|\delta_j^{(s)})^{y_{ij}} [1-P(q_k|\delta_j^{(s)})^{y_{ij}}]^{1-y_{ij}}}{\sum_{k'=1}^{K} \pi_{k'}^{(s)} \prod_{j}^{J} P(q_{k'}|\delta_j^{(s)})^{y_{ij}} [1-P(q_{k'}|\delta_j^{(s)})^{y_{ij}}]^{1-y_{ij}}}$$

a. Paso M.

i. Resolver el sistema de ecuaciones $\sum_{k=1}^{K} \frac{r_{jk}^{(s)} - n_k^{(s)} P(q_k|\delta_j^{(s)})}{[1-P(q_k|\delta_j^{(s)})]P(q_k|\delta_j^{(s)})} \frac{\partial P(q_k|\delta_j^{(s)})}{\partial \delta_{tj}^{(s)}} = 0$

$$\sum_{k=1}^{K} \frac{r_{jk}^{(s)} - n_k^{(s)} P(q_k|\delta_j^{(s)})}{[1-P(q_k|\delta_i^{(s)})]P(q_k|\delta_i^{(s)})} \frac{\partial P(q_k|\delta_j^{(s)})}{\partial \delta_{;i}^{(s)}} = 0$$

ii. Inicializar $m_0 = |\delta_1^{(s)} - \delta_1^{(s-1)}| m_0 = |\delta_1^{(s)} - \delta_1^{(s-1)}|$

iii. Para $j = 2, \ldots, J j = 2, \ldots, J$ hacer

1. $m_j = \max(m_{j-1}, |\delta_j^{(s)} - \delta_j^{(s-1)}|) m_j = \max(m_{j-1}, |\delta_j^{(s)} - \delta_j^{(s-1)}|)$

iv. $e = m_J e = m_J$

v. $\pi_k^{(s+1)} = n_k^{(s)} / N \pi_k^{(s+1)} = n_k^{(s)} / N$

El análisis del funcionamiento del algoritmo EM permitió identificar las dependencias que existen entre los valores que se calculan en cada uno de los pasos que se realizan en el proceso de estimación. Con esto se pueden identificar qué partes de este pueden ser paralelizables y cómo se dividirán los datos de entrada para su procesamiento. Se observa que en el paso E para obtener los estadísticos suficientes y completos es posible procesar cada fila de Y de manera individual debido a que no existen dependencias entre cada vector de respuestas observadas para obtener los resultados en el paso E. El proceso de suma de los resultados del cálculo para los N ítems es una operación que permite ser realizada sin importar el orden, por la propia propiedad asociativa de la misma. Con esto, el resultado de los estadísticos calculados se actualizaría en una variable global. Para la ejecución del paso M sucede lo anterior, ya que los parámetros pueden estimarse en una iteración para cada ítem de forma individual. Una vez estimados, se procede a calcular el conocimiento por cada fila de la matriz.

Solución aplicando computación de alto rendimiento

Para el desarrollo de la propuesta de solución se aplica la herramienta para el procesamiento *Big data Apache Spark* en su versión 2.1.1. Esto se realiza en conjunto con las herramientas Spring Boot 1.5 y Swagger 2 para la creación de una API REST que permita publicar servicios para estimar parámetros de los ítems a partir de un *dataset* y luego estimar conocimiento por cada estudiante a partir de las respuestas observadas.

A continuación, se muestra el esquema general de la propuesta de solución:

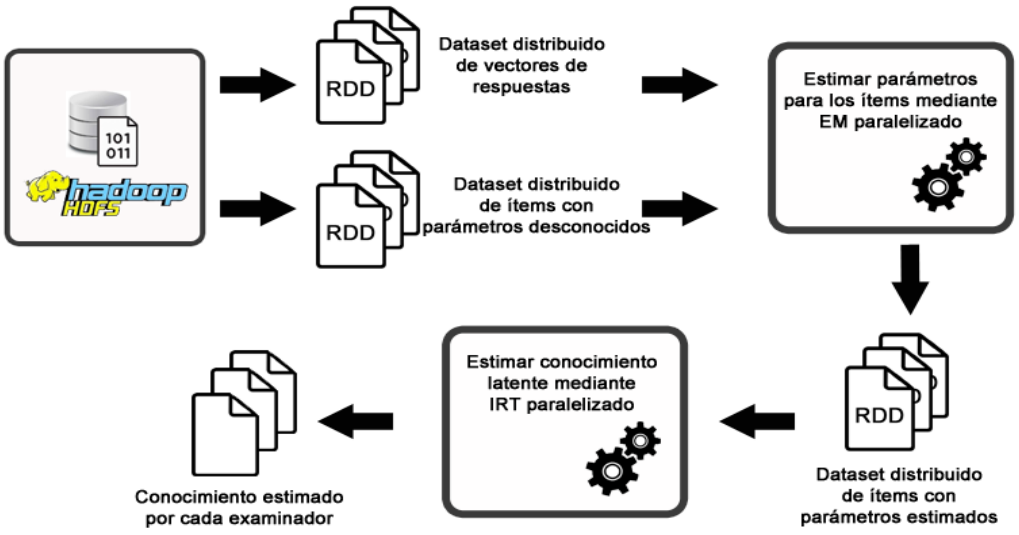

Figura 1. Esquema general de la propuesta de solución (Fuente: Elaboración propia)

El algoritmo se integrará a la plataforma de minado la cual es un clúster de computadoras desplegado en modo *Standalone*. Para su funcionamiento, los datos de entrada se toman de sistemas de archivos distribuidos HDFS que contienen los *datasets*. Se procede a estimar los parámetros de un ítem a partir de las respuestas observadas por los examinados y luego se estima el conocimiento latente para cada examinado. Ambos procesos se ejecutan de forma distribuida.

Como resultado de la ejecución del algoritmo se genera un fichero que se almacena en uno de los nodos físicos de la plataforma con los valores de estimación de conocimiento por cada estudiante y su correspondiente error estándar en la estimación.

Resultados y discusión

La adaptación del algoritmo IRT para estimar conocimiento latente sobre Datos Educacionales Masivos está accesible a través de una API de servicios REST. Con esta se puede acceder a los servicios de estimación de parámetros de un ítem y estimación de conocimiento para los examinados mediante el

envío de peticiones por el protocolo HTTP. Las respuestas del resultado del procesamiento efectuado por el clúster de computadoras son recibidas en formato JSON por el mismo protocolo. Este aporte práctico permitirá que otras plataformas educativas puedan consumir estos servicios disponibles y documentados para medir el conocimiento latente de los estudiantes en los cursos de forma global o por materias, conocer los parámetros de los ítems de sus instrumentos evaluativos en línea (discriminador, dificultad y probabilidad de ser contestados correctamente al azar) y predecir el éxito frente a un instrumento evaluativo.

A continuación, se presenta la vista general de la capa de servicios REST de la propuesta de solución:

Figura 2. Vista general de la capa de servicios REST de la propuesta de solución

Valoración del rendimiento computacional

Para la realización de las pruebas se rendimiento computacional se desplegó la propuesta de solución en un clúster de computadoras usando la herramienta *Apache Spark* en su versión 2.1.1. Se aplicó el modo *Standalone* de despliegue, donde se utilizan las funciones nativas de la herramienta. El clúster estaba conformado por una estación de trabajo que actuará como máster y cuatro estaciones de trabajo que actuarán como nodos trabajadores. Se debe tener en cuenta que las pruebas fueron realizadas en un entono no dedicado de estaciones de trabajo que pertenecen a la misma subred.

A continuación se muestran las especificaciones de *hardware* del clúster de computadoras empleados en las pruebas:

Tabla 2. Especificaciones de *hardware* del clúster

Estación de trabajo máster		
Tipo de procesador	**Cantidad de núcleos**	**Memoria principal**
Intel Celeron CPU G1839 2.8GHz	2	4GB DDR3
Estaciones de trabajo usadas como nodos trabajadores		
Tipo de procesador	**Cantidad de núcleos**	**Memoria principal**
Intel Core i3-4160 3.6GHz	4	4GB DDR3
Intel Celeron CPU G1839 2.8GHz	2	4GB DDR3
Intel Core i3-4160 3.6GHz	4	4GB DDR3
Intel Celeron CPU G1839 2.8GHz	2	4GB DDR3
Intel Core i3-4160 3.6GHz	4	4GB DDR3
Características de la red de datos		
Red cableada Fast Ethernet 100 Mbps		

Como especificación de software y sistema operativo instalado en las estaciones de trabajo empleadas en el clúster están las siguientes:

Tabla 3. Especificaciones de *software* del clúster

Sistema operativo
Ubuntu 16.04 LTS 64 bits, versión del núcleo: 4.4.0-121-generic
Software necesarios instalados
Máquina Virtual de Java OpenJDK 8 64bits, Versión de Java: 1.8.0_162
Scala 2.11.8
Apache Spark 2.1.1, versión de *Hadoop* 2.7.3

Para la ejecución de las pruebas se utilizaron *datasets* con vectores de respuestas dicotómicos de N filas por J columnas. Las filas representan el número de estudiantes y las columnas el número de ítems evaluativos. Se escalaron los datos de pruebas aumentando el número de estudiantes y los ítems. Se realizaron las mediciones incrementando además el número de procesadores para cada entrada de datos.

Se determinaron los siguientes requisitos para todas las mediciones realizadas:

- Para la estimación de parámetros se utilizó una distribución normal de 41 puntos y con el siguiente criterio de parada: error = 0.0001, iteraciones = 100.
- Para la estimación de conocimiento: error=0.0001, iteraciones=0.0001, conocimiento inicial = 1, umbral = 0.024.

A continuación, se muestran los resultados de las mediciones realizadas para *datasets* dicotómicos de diferentes tamaños escalando el número de procesadores de cuatro hasta dieciséis. En cada caso se utilizó 1GB de RAM por cada núcleo de procesamiento.

Tabla 4. Resultados por cada entorno de pruebas

Tamaño del *dataset* [KxN]: 1000x30		
Cantidad de procesadores	**Memoria utilizada (GB)**	**Tiempo paralelo (milisegundos)**
4	4	23020
8	8	19959
12	12	10396
Tamaño del *dataset* [KxN]: 1500x40		
Cantidad de procesadores	**Memoria utilizada (GB)**	**Tiempo paralelo (milisegundos)**
4	4	24747
8	8	20909
12	12	11360
Tamaño del *dataset* [KxN]: 2000x40		
Cantidad de procesadores	**Memoria utilizada (GB)**	**Tiempo paralelo (milisegundos)**
4	4	25688
8	8	21633
12	12	13541

Se realizaron pruebas para medir el tiempo de ejecución de la variante secuencial del algoritmo manteniéndose los mismos parámetros aplicados como condición de parada y los mismos juegos de datos. La medición se realizó ejecutando los mismos pasos de la propuesta de solución. Para la medición se utilizó una estación de trabajo con las mismas prestaciones del más potente de los esclavos.

Tabla 5. Datos de la estación de trabajo para pruebas secuenciales

Estación de trabajo usada para ejecutar el algoritmo secuencial		
Tipo de procesador	**Cantidad de núcleos**	**Memoria principal**
Intel Core i3-4160 3.6GHz	4	4GB DDR3

A continuación, se muestran los resultados de las mediciones:

Tabla 6. Resultados de las mediciones del algoritmo secuencial

Tamaño del *dataset* [KxN]	**Tiempo secuencial (milisegundos)**
1000x30	35916
1500x40	37341
2000x40	42916

Conclusiones

El estudio del modelo basado en Teoría de Respuesta al Ítem permitió caracterizar el proceso de estimación de conocimiento latente para un examinado. Con esto se identificaron los elementos necesarios que se necesitan para llevar a cabo este proceso. A partir de del cálculo de conocimiento latente por cada individuo se puede inferir la probabilidad de responder correctamente un ítem si se conocen sus parámetros.

El uso del algoritmo EM permitió aplicar un estimador basado en MML para estimar los parámetros de los ítems evaluativos a partir de las respuestas observadas, este proceso es fundamental para luego estimar el conocimiento latente por cada examinado.

El uso del paradigma *MapReduce* sobre la herramienta *Apache Spark* permitió obtener una solución escalable y tolerante a fallos que permite el procesamiento de datos educacionales masivos para estimar conocimiento latente.

Referencias

BAKER, F. B. (2001). *THE BASICS OF ITEM RESPONSE THEORY.* University of Wisconsin.

BAKER, R. S., & al., e. (2010). *Data mining for education* (Vol. 7). Oxford, UK: Elsevier Ltd.

Camargo-Vega, J. J., Camargo-Ortega, J. F., & Joyanes-Aguilar, L. (2015). Conociendo Big Data. *24*(38).

Coursera (2016). *How the world learn.* Coursera: http://coursera.tumblr.com/post/142363925112/introducing-our-new-infographic-how-the-world

Galindo, Á. J., & García, H. Á. (2010). *Minería de Datos en la Educación.* Madrid: Universidad Carlos III de Madrid.

Goldin, I. (2015). Move your lamp post: Recent data reflects learner knowledge better than older data. *7*(2).

Güler, N., Uyanık, G. K., & Teker, G. T. (2014). Comparison of classical test theory and item response theory in terms of item parameters. *2*(1).

Harwell, M. R. (1988). Item parameter estimation via marginal maximum likelihood and an EM algorithm: A didactic. *13*(3).

J. A. Larusson, B. W. (2014). *Learning Analytics. From research to practice.* United States.

JF, F., J, W., M, R., D, C., D, K., & E., M.-D. (2014). Item response theory, computerized adaptive testing, and PROMIS: assessment of physical function. *41*(1).

L. Thorpe, G., & Favia, A. (2012). *Data Analysis Using Item Response Theory Methodology: An Introduction to Selected Programs and Applications.* The University of Maine: DigitalCommons.

Manyika, J., Chui, M., & Brown, B. (Mayo de 2011). *Big data: The next frontier for innovation, competition, and productivity*. Obtenido de Digital Mckinsey: http://www.mckinsey.com/insights/business_technology/big_data_the_next_frontier_for_innovation

Melià, D. J. (2014). *Universidad de Valencia.* Recuperado el 1 de 1 de 2018, de Métodos de Estimación de Parámetros en Teoría de la Respuesta al Ítem: https://www.uv.es/~meliajl/Research/EstimationWeb/Fundamentos_archivos/frame.htm

Natarajan, D. V. (2009). *Basic Principles of IRT And Application to Practical Testing & Assessment.* Asilomar: MeritTrac Services (P) Ltd.

Pérez Gil, J. A. (2011). Modelos de Medición: Desarrollos actuales, supuestos, ventajas e inconvenientes. Universidad de Sevilla.

Romero, C., & Ventura, S. (2012). Data mining in education. *3*(1).

Romero, C., Ventura, S., Pechenizkiy, M., Baker, & S.J.d., R. (2011). *Handbook of Educational Data Mining.* New York: Taylor and Francis Group, LLC.

Sande, B. v. (2013). Properties of the Bayesian Knowledge Tracing Model. *5*(2).

Santos, R. P. (2015). Big Data: Philosophy, emergence, crowdledge, and science education. *8*(2).

Sehgal, G., & Garg, D. K. (2014). IMPROVED EXPECTATION MAXIMIZATION CLUSTERING ALGORITHM. *3*(5).

Yue, S., & K, H. R. (2017). Practical Consequences of Item Response Theory Model Misfit in the Context of Test Equating with Mixed-Format Test Data. *8*(484).