Optimización de algoritmos de repetición espaciada por comparación con calendarios de estudio por medio de algoritmos genéticos

Estanislao Claucich, Ramiro S. Garay y Axel A. Gorostidi
Universidad Nacional del Litoral. eclaucich@hotmail.com, ramiro.sgaray@gmail.com, gorounl@gmail.com

Resumen—El tiempo de estudio disponible para el aprendizaje es limitado; estudios anteriores mostraron que el aprendizaje guiado por máquina es capaz de mejorar la retención en memoria de los conocimientos adquiridos, empleando la misma cantidad de tiempo de estudio que otros métodos. Los algoritmos adaptativos que programan revisiones para maximizar la retención en memoria se conocen como algoritmos de repetición espaciada (SRS).

Se busca la optimización de los parámetros correspondientes a dos algoritmos SRS, uno de amplia utilización (SM2) y otro propuesto en trabajos anteriores, utilizando algoritmos genéticos y basándose en un dataset público que proviene de la plataforma de aprendizaje Duolingo.

La motivación de este estudio es aportar un método genérico para la obtención de los parámetros de funcionamiento de un algoritmo SRS, independientemente de su principio de funcionamiento, explotando la riqueza de datos disponibles de usuarios que utilizan plataformas virtuales de aprendizaje.

***Palabras clave*—SRS, calendarización, algoritmos genéticos, componente de conocimiento.**

I. Introducción

Para retener en memoria ciertos conocimientos, es sabido que uno debe estudiar la información en varias ocasiones. Es un resultado sabido de la psicología, que los tiempos en que se realizan estas revisiones afectan a la probabilidad de recordar ese conocimiento en el futuro [3].

Cepeda y otros mostraron en estudios hechos sobre estudiantes, que el intervalo de tiempo que hay entre el momento en que se aprende una colección de hechos, y el tiempo en que se los repasa, afecta de forma notable el rendimiento en una evaluación futura [4]. Esto lo demostraron para evaluaciones realizadas con varios intervalos de tiempo después de la adquisición de los conocimientos, demostrando su generalidad.

Esto se lo conoce como el efecto del espaciado (spacing effect) y constituye la base de los algoritmos de repetición espaciada, que intentan mejorar la retención del conocimiento espaciando los repasos de forma apropiada.

Los algoritmos SRS suelen basarse en distintos modelos de la memoria humana que describen su capacidad de retención del conocimiento en función del tiempo y los eventos que se consideren relevantes (como las actividades de repaso).

La curva que obtuvo Ebbinghaus se puede caracterizar como una de decaimiento exponencial, donde la probabilidad de recordar algún hecho disminuye exponencialmente con el paso del tiempo.

Este estudio plantea un algoritmo genético que permite la optimización de los parámetros que gobiernan el comportamiento de los SRS, basándose en la tasa de acierto media de los calendarios de estudio de un conjunto de usuarios de la plataforma de aprendizaje de idiomas Duolingo. El objetivo de este algoritmo es permitir adaptar genéricamente los parámetros de distintos SRS, tomando como medida de eficacia la capacidad del algoritmo SRS para producir calendarios similares a los de estudiantes de alta tasa de acierto.

Para ello, se probó el algoritmo sobre dos SRS: SuperMemo 2 (SM2) y un algoritmo de threshold creado para este estudio, basado en un modelo de memoria descrito y empleado por Lindsey y otros [2]. El algoritmo SM2 constituye un ejemplo interesante de SRS por su amplia utilización en programas de tarjetas electrónicas (flashcards) como son SuperMemo, Anki y Mnemosyne.

Por otro lado, el algoritmo de threshold competente a este trabajo se basa en un modelo de memoria más complejo y con más parámetros. Esto lo hace un buen candidato para ser entrenado por métodos evolutivos y probar su capacidad para producir calendarios de recomendaciones basándose en datos reales.

II. Conceptos

En este estudio, se denomina KC (knowledge component) a la unidad de conocimiento básica a memorizar. Un calendario de estudio es una serie de eventos de revisión y se encuentra asociado a algún usuario y KC particular. Todos los calendarios de estudios comienzan en el tiempo t = 0, que corresponde al tiempo de la primera revisión.

Cada revisión consiste en un número arbitrario de actividades que involucra recordar el KC estudiado. Por lo tanto, cada revisión tiene asociada una tasa de acierto que consiste del cociente a $= c / s$, donde $\boldsymbol{c}$ es el número de respuestas correctas, y $\boldsymbol{s}$ el número total de vistas.

Se evalúa la eficacia de un calendario en base a la tasa de acierto media, que es el promedio de la tasa de acierto sobre todos los eventos de revisión del calendario.

$$A = \sum_{i=1}^{N} \frac{C_i}{S_i} \quad (1)$$

Donde N es el número de eventos del calendario.

Esta medida se considera representativa de la calidad del espaciado de las revisiones, ya que una alta tasa de acierto implica que el evento de revisión anterior fue lo suficientemente reciente como para no equivocarse en el evento actual.

Un algoritmo SRS es capaz de programar una revisión en el futuro, dependiendo de ciertos parámetros internos que el algoritmo va adaptando dinámicamente dependiendo de la

evolución del usuario. De esta forma, intentan adaptar el espaciado de las repeticiones en base al rendimiento del usuario en la última revisión y todas las anteriores.

Dado un calendario real de N revisiones, correspondiente a un usuario y KC particular, es posible ejecutar un SRS sobre el mismo para obtener un calendario de recomendaciones. Este está constituido por los tiempos que el SRS sugiere para optimizar la retención del KC, a medida que se adapta a los eventos de revisión que el usuario fue tomando. Como la primera revisión puede ocurrir en cualquier momento, solo tiene sentido que contenga N-1 revisiones.

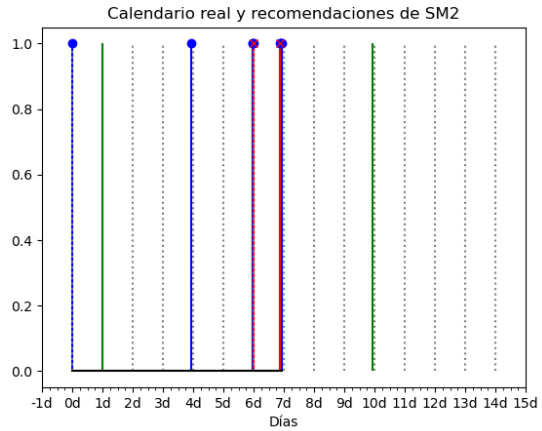

Fig. 1: En verde, los tallos representan el calendario de recomendaciones generado por el SM2 para el mismo calendario de estudio. Nótese que la primera recomendación se encuentra a 1 día de la primera revisión, y la siguiente a 6 días después de la segunda. Los tallos azules se corresponden con revisiones que no contienen errores, y los tallos rojos aquellas revisiones que contienen al menos un error.

Se define la distancia entre un calendario de estudio y uno de recomendaciones de la siguiente manera:

$$D = \frac{\Sigma_2^N |r_i - t_i|}{T} \tag{2}$$

Donde i es el número de revisión, T es el intervalo de tiempo total del calendario, r_i es la recomendación para la revisión i y t_i es el tiempo de revisión i.

Una distancia de 0 significa que el usuario siguió un calendario de estudio que concuerda perfectamente con las recomendaciones del SRS.

III. Materiales

A. Procesamiento de datos

El conjunto de datos original contiene aproximadamente 13 millones de entradas con un lapso de tiempo máximo de dos semanas cada una, pero a fines prácticos, se realizó un procesamiento de las mismas para obtener aquellas que son relevantes frente al modelo de estudio.

Esto se realiza ya que al ser datos reales de una plataforma digital en la cual cada usuario tiene la posibilidad de estudiar en el momento que desee, hay muchos calendarios que no contemplan un estudio consistente y prolongado a lo largo del tiempo. Por esto, se filtraron las entradas que contengan menos de 30 revisiones de un KC, dando como resultado un total de 6242 calendarios.

B. Dificultad de KC

Además, se computó para cada KC un índice de dificultad. Para realizar esto se sumaron, para cada KC, todas las veces totales que fue revisado, teniendo en cuenta todas las personas que lo estudiaron.

De forma similar se obtuvo el número de las veces totales que se repasó de forma acertada. La división de aciertos sobre totales da como resultado un número que expresa el porcentaje de aciertos del componente de estudio, el cual se relaciona con la "facilidad" del mismo.

Para expresar la "dificultad", se le resta a 1 la facilidad del KC en cuestión.

$$\delta = 1 - \frac{C_n}{N_n} \tag{3}$$

Donde δ refiere a la dificultad del KC, C_n a la suma total de todos los aciertos y N_n a las veces totales que se repasó.

IV. Métodos

A. Algoritmo de threshold

El SRS de threshold ideado para este estudio consta de varios parámetros y está basado en el modelo teórico-practico de memoria DASH, descrito por Lindsey y otros:

$$P_r(\alpha_s, \delta_i, \varphi, \omega) = logistic(\alpha_s - \delta_i + \boldsymbol{\varphi} \cdot \boldsymbol{C} - \boldsymbol{\psi} \cdot \boldsymbol{N})$$

$$C_i = \log(1 + c_{siw}), \quad 1 \le i \le 5 \tag{4}$$

$$N_i = \log(1 + n_{siw}), \quad 1 \le i \le 5$$

El modelo establece la probabilidad de que un individuo recuerde un KC en base a distintos parámetros. Tiene en cuenta la capacidad de aprendizaje del individuo (α_s), la dificultad del KC (δ_i) y dos vectores $\boldsymbol{\varphi}$ y $\boldsymbol{\omega}$ (pertenecientes a $\mathbb{R}^5$), asociados a ventanas de tiempo crecientes.

Estos ponderan la cantidad de veces que el individuo revisó (n_{siw}) y acertó en esas revisiones (c_{siw}) durante la ventana de tiempo w. Las ventanas de tiempo crecen exponencialmente y son solapadas. En este estudio se utilizaron 5 ventanas de tamaño {1,72; 2,95; 5,07; 8,72; 15} (medido en días).

El algoritmo de threshold propuesto emplea el modelo de memoria de Lindsey y un valor umbral de probabilidad para estimar el tiempo en que es necesario programar una revisión. De esta forma se mantiene la retención del KC con la potencia definida por el umbral.

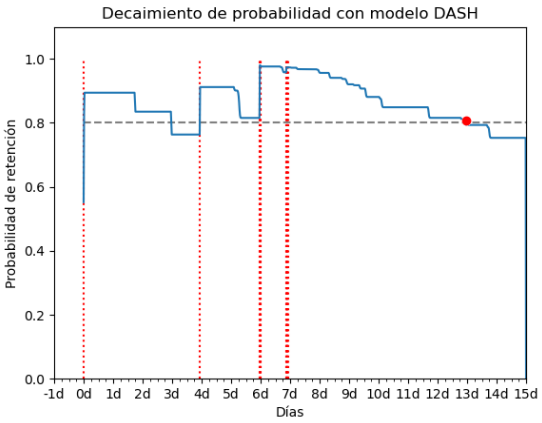

Fig. 2: Decaimiento de probabilidad de retención siguiendo el modelo DASH evolucionado. La última revisión se encuentra en el día 7, y el punto rojo indica la recomendación realizada por el modelo para un umbral de 80%.

B. Algoritmo SM2

SM2 es un algoritmo heurístico que emplea varios coeficientes en sus cálculos. En particular, emplea tres constantes en el cálculo de ajuste del EF (easiness factor):

$$E(n+1) = E(n) + \alpha - \beta.P(n) + \gamma.P(n)^2 \qquad (5)$$

Donde $P(n)$ es la tasa de aciertos del usuario en la revisión actual.

En la descripción original del algoritmo [5], las constantes toman los valores $\alpha = 0.1$, $\beta = -0.08$ y $\gamma = 0.02$. Para probar la capacidad del algoritmo genético para optimizar algoritmos SRS, estos tres parámetros fueron elegidos para ser evolucionados.

C. Algoritmo genético

Como ya se vio, tanto el modelo de DASH como el de SM2, utilizan una serie de parámetros constantes para realizar la predicción de la siguiente revisión. De esta forma, el algoritmo genético cuenta con individuos cuyo material genético se compone de estos parámetros de cada SRS, con el objetivo de encontrar aquellos valores que maximicen la eficiencia de estudio. Cada algoritmo SRS evoluciona por separado, permitiendo saber cuál de ellos es más capaz de emular los mejores calendarios de estudio del dataset.

Para poder determinar la eficiencia de un individuo (conjunto de parámetros particular) se eligen al azar N calendarios de estudio, y para cada uno de estos, se genera un calendario de recomendaciones. El fitness del individuo es la media de estos puntajes.

Este puntaje está formado por dos medidas, ***similitud*** y ***bondad***, las que determinarán la aptitud del individuo en base a los calendarios de recomendaciones que generó.

1. Función de similitud.

Por un lado, es de interés tener una medida de qué tan parecidas son, en función del tiempo, las revisiones del calendario de estudio, con respecto a las revisiones sugeridas por el calendario de recomendaciones.

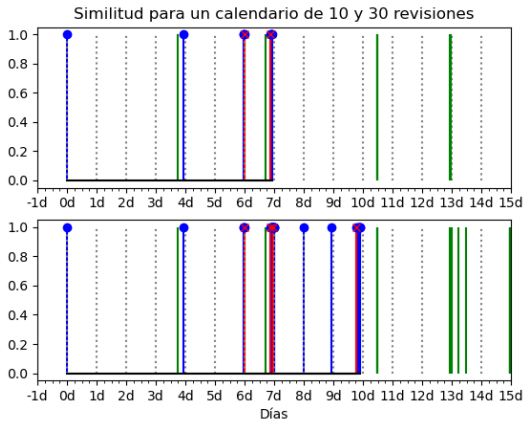

Fig. 3: Dos calendarios de estudio solapado con el calendario de recomendaciones generado por el modelo DASH evolucionado. El superior posee un valor de similitud de 0,0267 y el segundo 0,0074

Esta medida será función de la distancia entre dos calendarios, definida anteriormente. Donde a una distancia mínima igual a cero, la similitud es máxima. Y donde, a medida que la distancia aumenta, la similitud tienda a cero.

De esta manera, se obtiene una medida de qué tan parecido es el calendario que siguió realmente la persona con respecto al generado por el algoritmo genético.

$$S = \frac{e^{-kD}}{ts_n} \qquad (6)$$

Donde S es la similitud, D es la distancia entre dos calendarios, ts_n es la cantidad de revisiones y k una constante de suavidad.

2. Función de bondad.

Por otro lado, se quiere conocer no sólo el parecido entre los calendarios, sino también una medida que indique si el estudio realizado por la persona siguiendo dicho calendario, fue efectivo o no.

Esta medida de bondad, es simplemente la tasa de aciertos media de dicho calendario. Donde una bondad máxima de 1, se relaciona con un calendario donde no se cometieron errores, y una bondad mínima de 0, con un calendario donde no se cometieron aciertos.

Como se espera que la aptitud sea mayor para aquellos calendarios con una bondad cercana a 1. Se añade una penalización a los calendarios con una bondad relativamente baja.

Para esto, analizando el dataset, el promedio de la tasa de aciertos media de todos los calendarios disponibles es de 0,8843. De esta forma, todos los calendarios que posean una bondad menor a este promedio, serán considerados con una bondad igual a cero, dándole así, una mayor importancia a aquellos calendarios que se encuentran por encima de la media.

Entonces, a partir de estas dos medidas, se formuló una función de aptitud que determinará la eficiencia del calendario generado.

$$A = B.S\,, \quad tal\ que\ B = 0\ \forall B < 0{,}8843 \qquad (7)$$

Donde A es la aptitud, B es la bondad y S la similitud de un individuo en particular.

V. Resultados

Se realizaron tres pruebas distintas sobre la evolución de los algoritmos de DASH y SM2. Los parámetros que cambian entre las pruebas son la cantidad de calendarios de estudios ***N*** que se utilizan para la evaluación de cada agente, la cantidad de agentes en una población del algoritmo genético ***n***, y la cantidad máxima de generaciones ***gens*** durante la evolución (***Tabla 1***).

	N	n	Gens
1°	50	20	100
2°	100	30	50
3°	150	20	50

Tabla 1: Datos utilizados en cada una de las pruebas.

En cada prueba se utilizaron 5 particiones de los datos de entrada, con una distribución de 80/20 de datos de entrenamiento y prueba. A su vez, se utilizó elitismo y un 40% de brecha generacional en cada una de las pruebas.

A. Operador de cruza

Cada individuo se representa con un material genético de 30 bits, y se utiliza una cruza simple con un solo punto de corte, generando dos nuevos individuos en cada cruza realizada. Para cada prueba se utilizó una probabilidad de cruza del 90%.

B. Operador de mutación

Se utiliza un operador de mutación con un solo punto de mutación, donde toda la probabilidad de mutación recae sobre solo un alelo de todo el material genético, y la misma, para las tres pruebas, fue de 20%.

C. Operador de selección natural

Se utiliza una mejora de la selección natural por ruleta, donde el área de la misma se distribuye entre los individuos de la población en función de su aptitud normalizada. Esto reduce de manera considerable los problemas de mar de mediocres y mar de virtuosos [1].

DASH	Media	Varianza	Mediana
1°	0.61429	0,00643	0,61534
2°	0,61326	0,01303	0,62014
3°	0,60981	0,01577	0,61821

Tabla 2: Validación cruzada del modelo DASH.

DASH	α	Umbral
1°	0,22060	0,93719
2°	2,00950	0,93240
3°	1,08526	0,97053

DASH	φ_1	φ_2	φ_3	φ_4	φ_5
1°	-2,092	-4,332	2,352	-2,318	-4,788
2°	2,517	-4,292	-3,847	3,609	-0,065
3°	-1,628	2,193	4,624	-3,917	-3,831

DASH	ψ_1	ψ_2	ψ_3	ψ_4	ψ_5
1°	3,438	-4,192	0,782	-0,979	2,783
2°	4,714	0,169	1,056	3,729	4,817
3°	0,760	-1,697	1,560	3,181	-0,901

Tabla 3: Parámetros del mejor individuo en cada prueba.

SM2	Media	Varianza	Mediana	Mejora
Base	0,04870	0,00348	0,04765	-
1°	0,05382	0,00378	0,05293	10,5%
2°	0,05247	0,00313	0,05268	7,7%
3°	0.04950	0.00532	0.04713	1,6%

Tabla 4: Validación cruzada y mejora de las particiones realizadas sobre la evolución del modelo SM2.

VI. Conclusiones

Al existir una gran cantidad de parámetros a encontrar mediante la evolución, el espacio de búsqueda es relativamente grande, y se puede observar que no existe una alta relación entre los parámetros encontrados durante las 3 pruebas distintas en el modelo DASH. Sin embargo, las aptitudes medias de las pruebas, son muy similares entre sí.

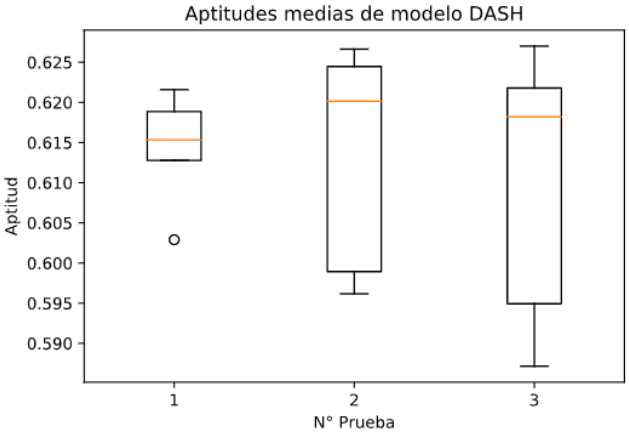

Fig. 4: Validación cruzada de las tres pruebas de modelo DASH

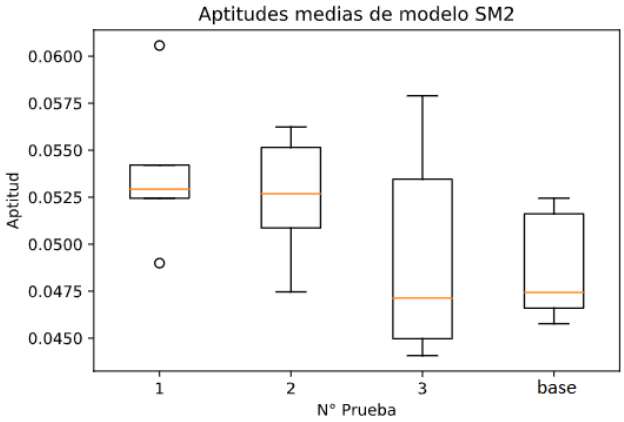

Fig. 5: Validación cruzada de las tres pruebas de modelo SM2

De esto se puede concluir que existen varios conjuntos de parámetros que llevan a comportamientos finales similares.

En el caso de la evolución del SM2, se observa que con la evolución de los parámetros se obtuvo una mejora máxima del 10,5% sobre la aptitud. Es decir, que existe un conjunto de parámetros que se adapta mejor a este modelo. Sin embargo, probablemente esta mejora no presente cambios considerables en la probabilidad de retención.

A su vez, se ve que la cantidad máxima de generaciones utilizada tiene una mayor influencia positiva en los resultados, que el aumento de la cantidad de calendarios de estudios ***N*** utilizados.

Referencias

[1] N. Razali y J. Geraghty, «Genetic Algorithm Performance with Different Selection Strategies in Solving TSP», ene. 2011, vol. 2.

[2] R. V. Lindsey, J. D. Shroyer, H. Pashler, y M. C. Mozer, «Improving students' long-term knowledge retention through personalized review», Psychol Sci, vol. 25, n.º 3, pp. 639-647, mar. 2014, doi: 10.1177/0956797613504302.

[3] N. J. Cepeda, E. Vul, D. Rohrer, J. T. Wixted, y H. Pashler, «Spacing effects in learning: a temporal ridgeline of optimal retention», Psychol Sci, vol. 19, n.º 11, pp. 1095-1102, nov. 2008, doi: 10.1111/j.1467-9280.2008.02209.x.

[4] J. T. Wixted y S. K. Carpenter, «The Wickelgren power law and the Ebbinghaus savings function», Psychol Sci, vol. 18, n.º 2, pp. 133-134, feb. 2007, doi: 10.1111/j.1467-9280.2007.01862.x.

[5] «SuperMemo.com». https://www.supermemo.com/en/archives1990-2015/english/ol/sm2 (accedido dic. 07, 2020).