Miscelánea Matemática **83** (2026) 45-53 SMM

DOI: https://doi.org/10.47234/mm.8303

Rudimentos de la escritura tensorial

Ricardo Abreu Blaya
Facultad de Matemática
Universidad Autónoma de Guerrero
Chilpancingo de los Bravo. México
rabreublaya@yahoo.es,

Ricardo Javier Abreu Angulo
Tecnológico Nacional de México
Chilpancingo de los Bravo. México
cayincode@gmail.com

y

José Luis Serrano Serrano
Ediciones Holguín
Holguín-Cuba
aneurisma71@gmail.com

1. Introducción

Todos los procesos escriturales funcionan a expensas de artificios que producen el sentido. Los estructuralistas han dejado claro que la forma produce al contenido. No existen contenidos netos, absolutos, independientes. El contenido no es una sustancia previa que, mediante determinadas operaciones, pueda ser vertida en un «recipiente» llamado comedia, ensayo, novela, soneto, romance, o cualquiera de los géneros literarios conocidos.

La situación es mucho más compleja en el caso particular de las escrituras poéticas. Al encontrarse menos dominados por la racionalidad, los discursos poéticos se constituyen en una zona de turbulencias. La imposibilidad de descifrar la naturaleza de los procesos que generan el sentido en los textos poéticos ha sido un quebradero de cabeza para hermeneutas y semiólogos.

Las vanguardias artísticas del siglo XX [2] subvirtieron radicalmente la relación entre contenido y forma. El dadaísmo, el surrealismo y, años más tarde, el Taller de Escritura Potencial (Oulipo, por su acrónimo en

Palabras clave: tensores, matrices circulantes, escrituras poéticas.

francés) [7] establecieron prácticas donde el problema de la emergencia del sentido fue colocado bajo el microscopio.

Mientras el surrealismo es antagónico a los principios lógicos de la razón y acude al inconsciente en la búsqueda de lo que denominan «libre ejercicio del pensamiento», el Oulipo traza la ruta en sentido contrario, elaborando un cúmulo de restricciones con la finalidad de propiciar la aparición del sentido. Dicho accionar lo desvincula, además, de los dadaístas y su culto al azar.

La idea central de la escritura tensorial puede resumirse en la aplicación de un operador multilineal que actúa sobre una constelación de objetos verbales (dominio) y devuelve un conjunto de nuevas constelaciones (imagen), cuyos contenidos son producidos automáticamente y trascienden la conciencia del propio escritor. Aunque las operaciones del tensor literario se encuentran claramente definidas, sus consecuencias semiológicas escapan al control de quien aporta el texto base sobre el cual actúa. Por supuesto, este tipo de escritura implica ciertas restricciones y por ello no se deslinda de la filosofía general del Oulipo.

En particular, cuando nos reducimos a la estructura del soneto, la escritura tensorial produce un conjunto de textos que claramente constituye un subconjunto de todas las posibles combinaciones producidas por intercambios entre los versos de los 14 sonetos que forman el dominio. En ese sentido, podemos decir que existe cierta afiliación con los Cien mil billones de poemas de Raymond Queneau [6]. Sin embargo, en este caso particular la escritura tensorial genera una cantidad finita (182) de nuevos sonetos cuyos contenidos son sustancialmente diferentes al de las estrofas originales. Esta operatoria queda estrictamente regulada por el operador multilineal definido.

2. Preliminares matemáticos

Los tensores son herramientas matemáticas fundamentales que permiten describir relaciones complejas en diversas disciplinas científicas. Son una generalización de los escalares, vectores y matrices, y se utilizan para representar magnitudes físicas que dependen del sistema de referencia, como la deformación de materiales, la curvatura del espacio-tiempo en la relatividad general y los estados cuánticos en la mecánica cuántica. Su capacidad para transformar información bajo cambios de coordenadas los hace esenciales en la física teórica y la ingeniería. Aunque su definición puede parecer abstracta, los tensores son la clave para comprender fenómenos que van desde la gravedad hasta el comportamiento de los materiales en condiciones extremas.

En este trabajo haremos un uso mucho menos tradicional de esta poderosa herramienta matemática. Comenzaremos por algunas definiciones necesarias para la comprensión de los resultados obtenidos. Con frecuencia seguiremos las notaciones empleadas en [1, 3].

Como es común, el espacio de matrices cuadradas de orden $n \times n$ con entradas en los reales, será denotado por $\mathbb{R}^{n\times n}$. De particular importancia serán las llamadas matrices circulantes:

Definición 2.1. Dado el vector columna

$$\mathbf{v} = \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix},$$

la matriz circulante generada por $\mathbf{v}$ se define por

$$\mathrm{circ}(\mathbf{v}) = \begin{pmatrix} v_1 & v_n & \cdots & v_2 \\ v_2 & v_1 & \cdots & v_3 \\ \vdots & \vdots & & \vdots \\ v_n & v_{n-1} & \cdots & v_1 \end{pmatrix}. \tag{1}$$

Un tensor de tercer orden suele representarse como una matriz multidimensional $\mathbb{A} \in \mathbb{R}^{n_1\times n_2\times n_3}$. Así, las matrices y los vectores pueden verse como tensores bidimensionales y unidimensionales, respectivamente. Para el propósito de este trabajo será suficiente restringir nuestro estudio al caso $n_1 = n_2 = n_3 = n$.

En particular, para $n = 3$ un tensor $\mathbb{A} \in \mathbb{R}^{3\times 3\times 3}$ puede ser visualizado como se muestra en la figura 1.

Para $\mathbb{A} \in \mathbb{R}^{n\times n\times n}$ se define la operación

$$\mathrm{unfold}(\mathbb{A}) = \begin{pmatrix} \mathcal{A}_1 \\ \mathcal{A}_2 \\ \vdots \\ \mathcal{A}_n \end{pmatrix},$$

donde cada $\mathcal{A}_j$ es una matriz de $\mathbb{R}^{n\times n}$ que puede ser vista como la «hoja» j-ésima del tensor $\mathbb{A}$.

De manera similar se define la operación inversa

$$\mathrm{fold}\begin{pmatrix} \mathcal{A}_1 \\ \mathcal{A}_2 \\ \vdots \\ \mathcal{A}_n \end{pmatrix} = \mathbb{A}$$

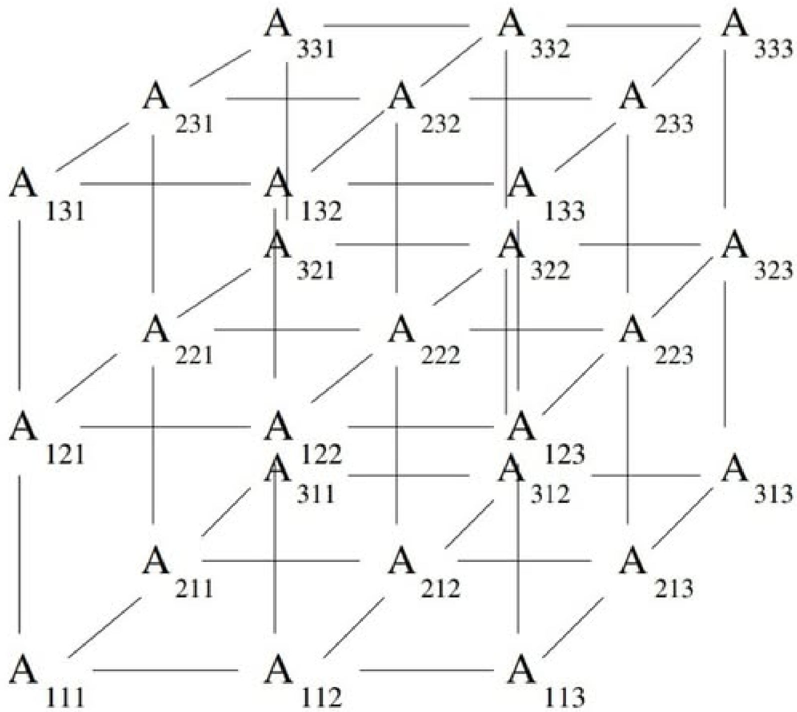

Figura 1. Tensor en $\mathbb{R}^{3\times 3\times 3}$.

3. Tensor literario

A partir de las matrices circulantes de $\mathbb{R}^{n\times n}$:

$$\mathcal{I}_1 = \operatorname{circ}\begin{pmatrix}1\\0\\\vdots\\0\end{pmatrix}, \mathcal{I}_2 = \operatorname{circ}\begin{pmatrix}0\\0\\\vdots\\1\end{pmatrix}, \ldots, \mathcal{I}_n = \operatorname{circ}\begin{pmatrix}0\\1\\\vdots\\0\end{pmatrix}$$

definamos el tensor

$$\mathbb{L} = \operatorname{fold}\begin{pmatrix}\mathcal{I}_1\\\mathcal{I}_2\\\vdots\\\mathcal{I}_n\end{pmatrix}. \tag{2}$$

Hacemos notar que la matriz $\mathcal{I}_1$ es precisamente la matriz identidad en $\mathbb{R}^{n\times n}$.

A continuación vamos a introducir un operador lineal $\mathcal{L}$ que actúa sobre el espacio de matrices cuadradas de orden $n \times n$. Para una matriz

$\mathcal{B} \in \mathbb{R}^{n\times n}$ se define

$$\mathcal{L} : \mathbb{R}^{n\times n} \mapsto \mathbb{R}^{n\times n}$$

según la fórmula:

$$\mathcal{L}[\mathcal{B}] := \text{fold}\left[(\text{unfold}(\mathbb{L}))^T \begin{pmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \vdots \\ \mathbf{b}_n \end{pmatrix}\right] = \text{fold}\begin{pmatrix} \mathcal{I}_1\mathbf{b}_1 \\ \mathcal{I}_2\mathbf{b}_2 \\ \vdots \\ \mathcal{I}_n\mathbf{b}_n \end{pmatrix},$$

donde $\mathbf{b}_1$, $\mathbf{b}_2$, $\ldots$, $\mathbf{b}_n$ son los vectores columna de $\mathcal{B}$.

Para una mejor comprensión de la forma en que actúa este operador tensorial sobre matrices cuadradas, ejemplificamos con el caso particular de $n = 3$.

Las matrices circulantes previamente definidas se escriben como

$$\mathcal{I}_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \mathcal{I}_2 = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}, \mathcal{I}_3 = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}.$$

Si $\mathcal{B}_0 \in \mathbb{R}^{3\times 3}$ tiene la forma

$$\mathcal{B}_0 = \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix},$$

entonces

$$\mathcal{L}[\mathcal{B}_0] = \text{fold}\begin{pmatrix} \mathcal{I}_1\mathbf{b}_1 \\ \mathcal{I}_2\mathbf{b}_2 \\ \mathcal{I}_3\mathbf{b}_3 \end{pmatrix} = \begin{pmatrix} b_{11} & b_{22} & b_{33} \\ b_{21} & b_{32} & b_{13} \\ b_{31} & b_{12} & b_{23} \end{pmatrix} =: \mathcal{B}_1.$$

Un cálculo directo permite comprobar que

$$\mathcal{L}^2[\mathcal{B}_0] := \mathcal{L}[\mathcal{B}_1] = \begin{pmatrix} b_{11} & b_{32} & b_{23} \\ b_{21} & b_{12} & b_{33} \\ b_{31} & b_{22} & b_{13} \end{pmatrix} =: \mathcal{B}_2$$

y finalmente que

$$\mathcal{L}^3[\mathcal{B}_0] := \mathcal{L}[\mathcal{B}_2] = \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix} = \mathcal{B}_0.$$

La siguiente proposición revela que la relación anterior es válida para cualquier dimensión n.

Proposición 3.1. *El operador $\mathcal{L}$ es una involución de grado n. Es decir, para toda $\mathcal{B} \in \mathbb{R}^{n\times n}$ se tiene*

$$\mathcal{L}^n[\mathcal{B}] := \underbrace{\mathcal{L}(\mathcal{L}(\cdots\mathcal{L}[\mathcal{B}]))}_{n-veces} = \mathcal{B}$$

Demostración. La prueba se basa en la siguiente propiedad cíclica que cumplen las matrices circulantes $\mathcal{I}_j \in \mathbb{R}^{n\times n}$ (véase por ejemplo [4, 5]):

$$\mathcal{I}_j^n := \underbrace{\mathcal{I}_j \cdot \mathcal{I}_j \cdots \mathcal{I}_j}_{n-veces} = \mathcal{I}_1, \text{ para todo } j = 1, \dots, n. \tag{3}$$

Efectivamente, un cálculo directo conduce a la expresión

$$\mathcal{L}^n[\mathcal{B}] = \text{fold}\begin{pmatrix} \mathcal{I}_1^n \mathbf{b}_1 \\ \mathcal{I}_2^n \mathbf{b}_2 \\ \vdots \\ \mathcal{I}_n^n \mathbf{b}_n \end{pmatrix}.$$

Luego de usar la propiedad (3), la fórmula anterior nos lleva a la identidad

$$\mathcal{L}^n[\mathcal{B}] = \text{fold}\begin{pmatrix} \mathcal{I}_1 \mathbf{b}_1 \\ \mathcal{I}_1 \mathbf{b}_2 \\ \vdots \\ \mathcal{I}_1 \mathbf{b}_n \end{pmatrix} = \text{fold}\begin{pmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \vdots \\ \mathbf{b}_n \end{pmatrix} = \mathcal{B},$$

con lo que se concluye la demostración. □

4. Aplicaciones

La transformación lineal descrita en la sección anterior está completamente caracterizada por el tensor $\mathbb{L}$. Este operador actúa sobre el espacio de matrices cuadradas de grado n, devolviendo una matriz perteneciente al mismo espacio. En este apartado proponemos hacer un uso iterado de dicha transformación para, a partir de un determinado conjunto de constelaciones verbales, generar un nuevo conjunto que no solo sobrepasa en número al conjunto original, sino que genera imágenes poéticas inesperadas y de sobrecogedora belleza.

Por brevedad, ilustraremos el proceso con el cuarteto, una estrofa de cuatro versos endecasílabos que riman en consonante según el esquema ABBA, lo que se conoce como rima abrazada. Es decir, el primer verso tiene rima consonante perfecta con el cuarto verso y esto mismo sucede entre el segundo y el tercer verso. Partiremos de un conjunto original de cuatro cuartetos (una matriz de dimensión 4×4) y a esta matriz le aplicaremos cuatro veces el operador tensorial $\mathcal{L}$ para de esta forma generar un conjunto de 16 cuartetos distintos. Debemos remarcar que esta operatoria se ha automatizado a través de AKIRA-01, un código en PYTHON que se describe en el Apéndice. Esto nos permite utilizar una técnica completamente análoga para composiciones literarias más extensas como, por ejemplo, la décima o el soneto, generando 100 décimas o 196 sonetos, según sea el caso, en un tiempo promedio de 0.130 segundos.

Ejemplo

Partimos de los siguientes cuartetos:

1. La víctima desciende la escalera / en busca del oscuro laberinto. / Hoy le toca morir al gallo pinto. / Estamos muertos de cualquier manera.
2. Dos borrachos caminan por la acera / tratando de que el lunes sea distinto. / Hay que olvidar el postulado quinto. / No se comprende el juego desde afuera.
3. El asesino puede ser cualquiera / con pasiones que nublan el instinto. / La locura dormita en su recinto. / La noche es una mala consejera.
4. El policía cruza la frontera / entre los cirios del placer extinto. / La sangre se parece al vino tinto. / Cualquier punto es el centro de la esfera.

Luego de correr AKIRA-01, se obtienen doce nuevos cuartetos:

1. La víctima desciende la escalera / tratando de que el lunes sea distinto. / La locura dormita en su recinto. / Cualquier punto es el centro de la esfera.
2. Dos borrachos caminan por la acera / con pasiones que nublan el instinto. / La sangre se parece al vino tinto. / Estamos muertos de cualquier manera.
3. El asesino puede ser cualquiera / entre los cirios del placer extinto. / Hoy le toca morir al gallo pinto. / No se comprende el juego desde afuera.
4. El policía cruza la frontera / en busca del oscuro laberinto. / Hay que olvidar el postulado quinto. / La noche es una mala consejera.
5. La víctima desciende la escalera / con pasiones que nublan el instinto. / Hoy le toca morir al gallo pinto. / La noche es una mala consejera.
6. Dos borrachos caminan por la acera / entre los cirios del placer extinto. / Hay que olvidar el postulado quinto. / Cualquier punto es el centro de la esfera.
7. El asesino puede ser cualquiera / en busca del oscuro laberinto. / La locura dormita en su recinto. / Estamos muertos de cualquier manera.
8. El policía cruza la frontera / tratando de que el lunes sea distinto. / La sangre se parece al vino tinto. / No se comprende el juego desde afuera.

9. La víctima desciende la escalera / entre los cirios del placer extinto. / La locura dormita en su recinto. / No se comprende el juego desde afuera.
10. Dos borrachos caminan por la acera / en busca del oscuro laberinto. / La sangre se parece al vino tinto. / La noche es una mala consejera.
11. El asesino puede ser cualquiera / tratando de que el lunes sea distinto. / Hoy le toca morir al gallo pinto. / Cualquier punto es el centro de la esfera.
12. El policía cruza la frontera / con pasiones que nublan el instinto. / Hay que olvidar el postulado quinto. / Estamos muertos de cualquier manera.

Conclusiones

En este trabajo se propuso una aplicación literaria de los tensores matemáticos. En particular, se construye un operador multilineal que permite producir nuevos textos, cuyos contenidos rebasan lo que puede generarse en la conciencia del propio escritor. Es importante señalar que la escritura tensorial coincide con el dadaísmo en la posibilidad de objetualizar el lenguaje y operar con las piezas resultantes en un juego de consecuencias imprevisibles. Sin embargo, a diferencia del reordenamiento azaroso de los recortes entreverados por el celebérrimo sombrero de Tristan Tzara, la actuación del tensor literario genera desplazamientos rigurosos en la masa textual. Si bien es cierto que la escritura tensorial produce resultados sorprendentes, su operatoria no es en ningún momento fortuita.

Bibliografía

[1] K. Braman, «Third-order tensors as linear operators on a space of matrices», *Linear Algebra and its Applications*, vol. 433, 2010, 1241–1253, https://doi.org/10.1016/j.laa.2010.05.025.

[2] M. de Micheli, *Las vanguardias artísticas del siglo XX*, Ediciones Unión, La Habana, Cuba, 1967.

[3] M. Kilmer, C. Martin y L. Perrone, *A third-order Generalization of the Matrix SVD as a Product of Third-order Tensors*, Tech. Rep. TR-2008-4, Tufts University, Department of Computer Science, October 2008.

[4] I. Kra y S. R. Simanca, «On circulant matrices», *Notices of AMS*, vol. 59, núm. 3, 2012, 568–377, http://dx.doi.org/10.1090/noti804.

[5] T. Lara, «Matrices circulantes», *Divulgaciones Matemáticas*, vol. 9, núm. 1, 2001, 85–102.

[6] R. Queneau, *Cent mille milliards de poèmes*, Edité par Nrf Gallimard, 1961.

[7] R. Queneau, G. Perec, F. L. Lionnais, I. Calvino y H. Mathews, *Ejercicios de literatura potencial*, Caja Negra Editora, Buenos Aires, Argentina, 2016.

5. Apéndice: AKIRA-01

```
filename="example.txt"
def length(file):
    with open(file, 'r') as txt:
        content=txt.read()
counter=0
    for line in content.splitlines():
        line = line.strip()
if line: counter+=1
        else: break
return counter
def sorting(file,iterator):
    if iterator==0:
        with open(file, 'r') as txt: content=txt.read()
    else:
        with open(file.rstrip(".txt")+str(iterator+1)+".txt", 'r') as txt:
            content=txt.read()
sublist=[]; matrix=[]
    for line in content.splitlines():
        line = line.strip()

        if line: sublist.append(line)
        elif sublist: matrix.append(sublist); sublist = []

    if sublist: matrix.append(sublist)
matrix_length=len(matrix)
    new_matrix = [["" for _ in range(matrix_length)]
  for _ in range(matrix_length)]

    for j in range(0,matrix_length):
        for k in range(0,matrix_length):

            if j-k>=0: new_matrix[j-k][k] = matrix[j][k]
            else: new_matrix[j+(matrix_length-k)][k] = matrix[j][k]
return new_matrix
for i in range(length(filename)):
    output=open((filename.strip(".txt")+str(i+2))+".txt","a")
for j in sorting(filename,i):
        for k in j:
output.write(k + "\n")
        output.write("\n")
```