Cálculo privado de la distancia de Wasserstein (Earth Mover)

Alberto Blanco-Justicia
Universitat Rovira i Virgili
Departament d'Enginyeria Informàtica i Matemàtiques
Av. Països Catalans 26, Tarragona
alberto.blanco@urv.cat

Josep Domingo-Ferrer
Universitat Rovira i Virgili
Departament d'Enginyeria Informàtica i Matemàtiques
Av. Països Catalans 26, Tarragona
josep.domingo@urv.cat

***Resumen*—La distancia de Wasserstein, más conocida en inglés como *Earth Mover's Distance* (EMD), es una medida de distancia entre dos distribuciones de probabilidad. La EMD se utiliza ampliamente en la comparación de imágenes y documentos, y forma parte de modelos de privacidad como la t-proximidad. En este artículo, primero mostramos que para distribuciones discretas unidimensionales esta métrica se puede reducir al cálculo del tamaño de la intersección de dos conjuntos. Luego usamos esquemas de intersección segura de conjuntos para construir un mecanismo de cálculo privado de la EMD: dos propietarios de ficheros privados pueden calcular la EMD entre sus ficheros respectivos sin que ningún propietario deba revelar su fichero al otro. Demostramos el funcionamiento de nuestra propuesta mediante un servicio de búsqueda inversa de imágenes cifradas almacenadas en un servidor externo.**

***Palabras clave*—Computación multiparte segura, intersección segura de conjuntos, cifrado buscable.**

I. Introducción

La distancia de Wasserstein, más conocida con el término inglés de *Earth Mover's Distance* (EMD), es una medida de distancia entre distribuciones que mide el coste mínimo necesario para transformar una distribución en otra. Imaginando las distribuciones como dos montones de tierra, y de ahí el nombre de la distancia, el coste de transformar un montón de tierra en otro es igual a la cantidad de tierra a mover multiplicada por la distancia que hay que moverla [1]. La EMD es un caso especial de un problema de optimización de transporte y, para distribuciones unidimensionales y discretas (que puedan expresarse como un histograma), existen algoritmos eficientes para calcularla.

La EMD resulta útil para comparar imágenes en términos de su distribución de colores y texturas [1], [2], y para comparar documentos semánticamente [3], [4]. Por ello puede usarse para implementar bases de datos de imágenes o documentos con soporte para búsqueda inversa. La EMD se ha utilizado también en la anonimización de microdatos: el modelo de privacidad de la t-proximidad [5], extensión del k-anonimato [6], requiere que los atributos confidenciales en cada clase k-anónima tengan una distribución próxima a la distribución de dichos atributos confidenciales en todo el fichero. Esta proximidad se mide utilizando la EMD.

En este trabajo proponemos un mecanismo para el cálculo privado de la EMD. Dos individuos, cada uno de ellos poseedor de un fichero, pueden calcular la EMD entre sus ficheros sin que ninguno de los dos deba revelar su fichero a la otra parte. El mecanismo que proponemos está basado en el cálculo seguro biparte del tamaño de la intersección de conjuntos [7]. Nuestro mecanismo puede servir como base para la implementación de cifrado buscable (*searchable encryption*) para búsquedas inversas de imágenes y documentos o para el desarrollo de mecanismos distribuidos de anonimización.

En la sección II demostramos que para distribuciones unidimensionales discretas, el cálculo de la EMD puede reducirse al cálculo de la cardinalidad de la intersección entre dos conjuntos. La sección III describe métodos para el cálculo privado de la cardinalidad de la intersección de dos conjuntos. La sección IV presenta nuestro trabajo experimental, en el que aplicamos nuestro método de cálculo de la EMD a un sistema de búsqueda inversa de imágenes cifradas almacenadas en un servidor externo. En la sección V resumimos la conclusiones del artículo.

II. La distancia de Wasserstein

Dadas dos distribuciones de probabilidad A y B, la EMD mide la distancia entre A y B [1]. En el caso general, la EMD se obtiene de resolver un problema de optimización de transporte. En el caso particular de distribuciones discretas unidimensionales o histogramas de frecuencias relativas $A = \{a_0, \ldots, a_{n-1}\}$ y $B = \{b_0, \ldots, b_{n-1}\}$, la EMD se puede calcular mediante el siguiente algoritmo iterativo:

Algoritmo 1: Distancia de Wasserstein (EMD) entre dos distribuciones unidimensionales discretas

Input: $A = \{a_0, \ldots, a_{n-1}\}$, $B = \{b_0, \ldots, b_{n-1}\}$
Output: $\mathrm{EMD}(A, B)$

$w_0 = 0$
for $i = 1 \to n$ **do**
  $w_i = a_{i-1} - b_{i-1} + w_{i-1}$
end
return $\sum_{i=0}^{n} |w_i|$

A continuación mostraremos cómo el cálculo de la EMD entre dos distribuciones unidimensionales discretas A y B se puede reducir a operaciones entre la cardinalidad de dos conjuntos que codifican estas distribuciones y la cardinalidad de su intersección. Esta reducción nos permitirá obtener un protocolo para el cálculo seguro biparte de la EMD. Para ello, primero demostraremos que el algoritmo anterior es equivalente a la distancia de Manhattan entre distribuciones acumuladas.

Teorema 1. *Sea $\tilde{A}$ (resp. $\tilde{B}$) la función de distribución acumulada o suma acumulativa de A (resp. B). Entonces la*

distancia $EMD(A,B)$ puede calcularse como la norma-1, o distancia de Manhattan, de $\tilde{A}$ y $\tilde{B}$:

$$EMD(A,B) = \|\tilde{A} - \tilde{B}\|_1. \quad (1)$$

Demostración. Dada la distribución discreta unidimensional $A = \{a_0, \ldots, a_{n-1}\}$, la distribución acumulada de A es

$$\tilde{A} = \{\tilde{a}_i = \textstyle\sum_{j=0}^{i} a_i \ : \ 0 \le i \le n-1\}. \quad (2)$$

Por otra parte, la norma-1 o distancia de Manhattan se define como

$$\|A - B\|_1 = \textstyle\sum_{i=0}^{n-1} |a_i - b_i|. \quad (3)$$

Desarrollando la línea 3 del Algoritmo 1, referente al cálculo de w_i, y reordenando los factores obtenemos, para $i > 0$ y siendo $w_0 = 0$,

$$\begin{aligned} w_i &= a_{i-1} - b_{i-1} + w_{i-1} \\ &= a_{i-1} - b_{i-1} + a_{i-2} - b_{i-2} + w_{i-2} \\ &\cdots \\ &= a_{i-1} - b_{i-1} + a_{i-2} - b_{i-2} + \cdots + a_0 - b_0 \\ &= (a_{i-1} + \cdots + a_0) - (b_{i-1} + \cdots + b_0) \\ &= \textstyle\sum_{j=0}^{i-1} a_j - \sum_{j=0}^{i-1} b_j. \end{aligned} \quad (4)$$

Según la línea 5 del Algoritmo 1 y aplicando las ecuaciones (2), (3) y (4), obtenemos

$$\begin{aligned} EMD(A,B) &= \textstyle\sum_{i=0}^{n} |w_i| \\ &= w_0 + \textstyle\sum_{i=1}^{n} |w_i| \\ &= \textstyle\sum_{i=1}^{n} |w_i| \\ &\overset{(4)}{=} \textstyle\sum_{i=1}^{n} \left| \sum_{j=0}^{i-1} a_j - \sum_{j=0}^{i-1} b_j \right| \\ &\overset{(2)}{=} \textstyle\sum_{i=1}^{n} |\tilde{a}_{i-1} - \tilde{b}_{i-1}| \\ &= \textstyle\sum_{i=0}^{n-1} |\tilde{a}_i - \tilde{b}_i| \\ &\overset{(3)}{=} \|\tilde{A} - \tilde{B}\|_1. \end{aligned}$$

□

Ahora nos falta demostrar que la distancia de Manhattan puede obtenerse del cálculo de la cardinalidad de la intersección de dos conjuntos. Este resultado proviene de [8] y lo desarrollamos a continuación.

Primero, definimos una función de codificación f que toma una lista $A = \{a_0, \ldots, a_{n-1}\}$ de enteros no negativos y devuelve un conjunto de la siguiente forma:

$$\mathcal{A} = f(A) = \{(i,j) : a_i > 0, \ 1 \le j \le a_i\}$$

Por ejemplo, dada una lista $A = \{2,1,3\}$, el conjunto resultante $\mathcal{A}$ es $\{(1,1),(1,2),(2,1),(3,1),(3,2),(3,3)\}$.

Teorema 2. *Dada la función de codificación f, las listas A y B de valores no negativos y del mismo tamaño n, y sus respectivas codificaciones $\mathcal{A}$ y $\mathcal{B}$, se cumple*

$$\|A - B\|_1 = |\mathcal{A}| + |\mathcal{B}| - 2|\mathcal{A} \cap \mathcal{B}|.$$

Demostración. Primero, reescribimos la ecuación para la distancia de Manhattan de las listas A y B como la diferencia entre la suma de los valores máximos par a par menos la suma de los valores mínimos par a par de A y B:

$$\begin{aligned} \|A - B\|_1 &= \textstyle\sum_{i=0}^{n-1} |a_i - b_i| \\ &= \textstyle\sum_{i=0}^{n-1} \max(a_i, b_i) - \min(a_i, b_i) \\ &= \textstyle\sum_{i=0}^{n-1} \max(a_i, b_i) - \sum_{i=0}^{n-1} \min(a_i, b_i). \end{aligned}$$

Observamos, además, que la cardinalidad del conjunto resultante de aplicar la función de codificación f a una lista es igual a la suma de los valores de dicha lista:

$$|\mathcal{A}| = \textstyle\sum_{i=0}^{n-1} a_i.$$

La función de codificación f tiene además la siguiente propiedad: el tamaño de la unión ente las codificaciones de A y B es la suma de los valores máximos, par a par, de A y B:

$$\begin{aligned} |\mathcal{A} \cup \mathcal{B}| &= |\{(i,j) : (i,j) \in \mathcal{A} \text{ o } (i,j) \in \mathcal{B}\}| \\ &= |\{(i,j) : a_i, b_i > 0, \ 1 \le j \le \max(a_i, b_i)\}| \\ &= \textstyle\sum_{i=0}^{n-1} \max(a_i, b_i). \end{aligned}$$

Del mismo modo, el tamaño de la intersección es igual a la suma de los valores mínimos.

$$\begin{aligned} |\mathcal{A} \cap \mathcal{B}| &= |\{(i,j) : (i,j) \in \mathcal{A} \text{ y } (i,j) \in \mathcal{B}\}| \\ &= |\{(i,j) : a_i, b_i > 0, \ 1 \le j \le \min(a_i, b_i)\}| \\ &= \textstyle\sum_{i=0}^{n-1} \min(a_i, b_i). \end{aligned}$$

Así, la distancia de Manhattan entre A y B resulta de la diferencia entre las cardinalidades de la unión y la intersección de los conjuntos resultantes de aplicar la función f a A y B. Dada la identidad $|A \cup B| = |A| + |B| - |A \cap B|$, finalmente nos queda

$$\|A - B\|_1 = |\mathcal{A}| + |\mathcal{B}| - 2|\mathcal{A} \cap \mathcal{B}|.$$

□

De los Teoremas 1 y 2 obtenemos, finalmente, que el cálculo de la EMD se puede expresar como

$$EMD(A,B) = |\tilde{\mathcal{A}}| + |\tilde{\mathcal{B}}| - 2|\tilde{\mathcal{A}} \cap \tilde{\mathcal{B}}|.$$

II-A. Expansión de los mensajes

Sea el tamaño del conjunto $|A| = n$ y su suma $s = \sum_{i=0}^{n-1} a_i$. Entonces la expansión del mensaje obtenido al calcular la suma acumulativa $\tilde{A}$ de A y codificarla mediante la función f es:

- En el mejor caso, cuando $a_0 = \cdots = a_{n-2} = 0$ y $a_{n-1} = s$, $|\tilde{\mathcal{A}}| = s$.
- En el peor caso, cuando $a_0 = s$ y $a_1, \cdots, a_{n-1} = 0$, $|\tilde{\mathcal{A}}| = sn$.
- En el caso medio $|\tilde{\mathcal{A}}| = (1+n)s/2$.

Así, la complejidad espacial de nuestra propuesta es $\mathcal{O}(sn)$.

III. Cálculo privado de la cardinalidad de la intersección de conjuntos

Habiendo reducido el cálculo de la EMD al cálculo de la cardinalidad de la intersección de conjuntos, podemos obtener un protocolo para el cálculo de la EMD partiendo de mecanismos existentes para el cálculo seguro multiparte de esta operación. Estos mecanismos se denominan intersección privada de conjuntos-cardinalidad (*Private Set Intersection – Cardinality*, PSI-CA), y son una primitiva recurrente en los protocolos de computación multiparte en minería de datos privados. Su principal uso es el de compartir datos entre entidades que no confían plenamente entre ellas. En lugar de compartir todos los datos que cada una de ellas almacena, pueden comenzar por intercambiar información sobre datos que tienen en común. Un ejemplo de ello podría ser la búsqueda de movimientos fraudulentos llevados a cabo por clientes de entidades financieras. Otros usos de los protocolos de PSI-CA incluyen la búsqueda de documentos por palabras clave en bases de datos externas o el emparejamiento de usuarios en redes sociales según los contactos que comparten.

Uno de los trabajos sobre PSI-CA más influyentes es el de Freedman, Nissim y Pinkas [7]. Este trabajo incluye distintos protocolos para calcular diferentes operaciones entre conjuntos de forma segura, incluyendo el tamaño de la intersección. Todas ellas se basan en la misma primitiva, la evaluación insconsciente de polinomios (*oblivious polynomial evaluation*, OPE). Una de las partes construye un polinomio cuyas raíces son los elementos de su conjunto. La otra parte evalúa este polinomio para todos los elementos de su conjunto. Aquellos elementos cuya evaluación dé 0 son elementos compartidos por ambas partes. Mediante esquemas de cifrado homomórfico para la suma (por ejemplo, Paillier [9]), la segunda parte puede evaluar el polinomio aunque solamente conozca sus coeficientes cifrados. Por desgracia, estos protocolos requieren el cálculo de operaciones criptográficas poco eficientes y no están, por lo tanto, indicados para grandes conjuntos de datos, aunque [10] presenta optimizaciones.

Otra construcción ampliamente utilizada en protocolos de PSI-CA son los filtros de Bloom. Dong, Chen y Wen [11], utilizan una modificación de los filtros de Bloom y protocolos de transferencia inconsciente para calcular operaciones entre multi-conjuntos. Pinkas, Schneider y Zohner [12] propusieron una mejora en la escalabilidad del protocolo.

III-A. PSI-CA basado en la evaluación inconsciente de polinomios

A continuación describimos brevemente el cálculo del tamaño de la intersección de dos conjuntos mediante la evaluación inconsciente de polinomios. Sean $\mathcal{C}$ y $\mathcal{S}$ los dos participantes del protocolo, con $\mathcal{C}$ aportando el conjunto $A = \{a_0, \ldots, a_{n-1}\}$ y $\mathcal{S}$ el conjunto $B = \{b_0, \ldots, b_{m-1}\}$. Enc representa aquí cifrado con el criptosistema de Paillier bajo la clave pública de $\mathcal{C}$. El criptosistema de Paillier es homomórfico para la suma y cumple las siguientes propiedades:

$$Enc(x) \cdot Enc(y) = Enc(x + y)$$
$$Enc(x)^k = Enc(kx)$$

Protocolo 1. *PSI-CA-OPE*

1. *$\mathcal{C}$ calcula* $P(x) = \sum_{i=0}^{n} p_i x^i = \prod_{j=0}^{n-1}(x - a_i)$.
2. *$\mathcal{C}$ envía* $Enc(p_0), \ldots, Enc(p_n)$ *a $\mathcal{S}$.*
3. *$\mathcal{S}$ genera un número aleatorio* $r_j \in \mathbb{Z}_n$ *para cada* $1 \leq j \leq m-1$. *$\mathcal{S}$ calcula* $Enc(r_j \cdot p(b_j) + k)$ *para cada* $1 \leq j \leq |B|$. *A continuación, $\mathcal{S}$ devuelve estos textos cifrados a $\mathcal{C}$.*
4. *$\mathcal{C}$ descifra los valores recibidos. El resultado de cada descifrado es k o un elemento aleatorio.*
5. *El número de k's recibidas es igual a* $|A \cap B|$.

Este protocolo es seguro en presencia de adversarios semihonestos. Para que $\mathcal{C}$ y $\mathcal{S}$ obtengan los resultados, este protocolo se debe ejecutar dos veces, con $\mathcal{C}$ y $\mathcal{S}$ intercambiando papeles. Tras ejecutar este protocolo, $\mathcal{C}$ y $\mathcal{S}$ obtienen $|A|$, por el grado del polinomio, $|B|$, por el número de elementos devueltos en el paso 3, y $|A \cap B|$. Estos tres elementos son suficientes para calcular la EMD entre A y B.

III-B. PSI-CA basado en filtros de Bloom

En esta sección describimos las propiedades de los filtros de Bloom que permiten su uso como base para el desarrollo de protocolos de PSI-CA. Los filtros de Bloom [13] son una estructura de datos probabilística para codificar conjuntos que permite consultas de membresía. Las consultas de membresía para elementos originalmente miembros del conjunto devolverán cierto en el $100\,\%$ de los casos, mientras que para elementos que no forman parte del conjunto existe cierta probabilidad de falso positivo.

Un filtro de Bloom es un vector $B \in \mathbb{Z}_2^m$, con todos sus elementos inicializados a 0, y acompañado por $k \ll m$ funciones de hash $H_i : \{0,1\}^* \to \{0, \ldots, m-1\}$. Para insertar un elemento e, se calculan los índices $(i_0, \ldots, i_{k-1}) = (H_0(e), \ldots, H_{k-1}(e))$ y se asigna un 1 en las posiciones correspondientes del vector. Del mismo modo, una consulta de membresía para el elemento e se resuelve comprobando si sus k posiciones en el vector ya valen 1. La probabilidad de falso positivo para las consultas de membresía en un filtro de Bloom que contenga n elementos viene dada por la expresión $(1-(1-\frac{1}{m})^{kn})^k$, que es la probabilidad de que los k índices del elemento que se busca sean iguales a 1 aunque el elemento no esté en el filtro. Ello puede ocurrir porque los índices para otro elemento insertado coincidan con los del elemento que se busca (muy poco probable si usamos funciones de hash criptográficas) o bien con los de una combinación de otros elementos insertados.

El número aproximado de elementos insertados en un filtro de Bloom es

$$|A| \approx -\frac{m}{k} \ln\left(1 - \frac{\mathcal{H}(B_A)}{m}\right). \tag{5}$$

donde A es el conjunto codificado, B_A es la codificación de A, $\mathcal{H}(\cdot)$ es el peso de Hamming, m es la longitud del filtro de Bloom y k es el número de funciones de hash.

El filtro de Bloom que codifica la unión o intersección de dos conjuntos codificados se puede calcular fácilmente aplicando las operaciones $\vee$ ("o" lógica) y $\wedge$ ("y" lógica) bit a bit, respectivamente. Aplicando la Expresión (5) al filtro de Bloom resultante, podremos obtener estimaciones de los cardinales de la unión y la intersección de los conjuntos.

Los filtros de Bloom, por sí solos, no ofrecen suficiente protección a los elementos codificados como para ser aplicados

directamente a PSI-CA. Si ambas partes comparten un universo de elementos, cualquiera de ellas podría obtener todos los elementos del conjunto de la otra parte mediante fuerza bruta. Mediante protocolos de transferencia inconsciente, una de las partes puede pedir a la otra todos los índices para los que su filtro de Bloom es igual a 1 y así obtener la intersección de los conjuntos [11]. Otra posibilidad es cifrar cada posición del filtro de Bloom con criptosistemas homomórficos para la suma binaria, como Goldwasser-Micali, y sumar ambos filtros. Las posiciones donde ambos filtros coinciden darán 0 y el resto 1 [14]. Otra aproximación consiste en alterar ciertas posiciones del filtro de Bloom con cierta probabilidad, como en la respuesta aleatorizada [15]. Finalmente, si utilizamos funciones de hash con clave, como HMAC, sólo aquellos que conozcan la clave pueden insertar elementos y hacer consultas, lo que permite construir esquemas de cifrado buscable de clave simétrica, como hacemos en nuestros experimentos.

IV. Experimentos

En nuestro trabajo experimental simulamos un sistema de búsqueda inversa de imágenes cifradas almacenadas en un servidor externo. La implementación de los experimentos, en forma de Jupyter Notebook, se encuentra en Github[1]. Las imágenes para el experimento están extraídas de [16], que contiene 9144 imágenes de 101 categorías distintas.

Tomamos 1024 de esas imágenes para construir nuestro índice. Para cada imagen, obtenemos su histograma de grises, con $n = 16$ intérvalos de 16 valores cada uno y lo reescalamos para que su suma acumulada sea $s = 100$. A continuación calculamos el histograma acumulado y lo codificamos mediante la función f. Los elementos resultantes se incluyen en un filtro de Bloom con $k = 4$ hashes y tamaño $m = 8000$. La implementación de los filtros de Bloom usa funciones de hash con clave secreta (HMAC), de modo que sólo quien posee la clave secreta es capaz de añadir imágenes a la base de datos y de hacer búsquedas en ella. Esto nos da un índice de $7,8MB$. El tiempo de construcción de los índices es de $29,86$ minutos, a $1,75$ segundos por imagen.

Para efectuar búsquedas, la imagen a buscar se codifica de la misma manera que las imágenes del índice y se calcula su EMD respecto a cada uno de los elementos del índice. Todas las imágenes cuya distancia se encuentre por debajo de un umbral preestablecido se devuelven como resultado. En nuestros experimentos, el umbral se ha fijado en 10. El tiempo de búsqueda es de 31 milisegundos.

Se espera cierto error en los cálculos de la EMD. Por una parte, exigimos que todos los histogramas de grises tengan la misma suma acumulada y solamente podemos operar con enteros. Por tanto, los redondeos introducirán cierto error. Por otra parte, los filtros de Bloom tienen cierta probabilidad, aunque pequeña, de devolver falsos positivos, y esa posibilidad se presenta también cuando calculamos las intersecciones. En nuestros experimentos, con 1024 imágenes, el error absoluto medio entre el cálculo de la EMD en claro y el cálculo usando nuestro mecanismo es $MAE = 4,81$. Las distancias entre imágenes, calculadas par a par en claro y utilizando nuestro método se muestran como matrices de distancias en las Figuras 1 y 2.

[1] https://github.com/ablancoj/privateEMD/blob/master/EMD%20Bloom%20images.ipynb

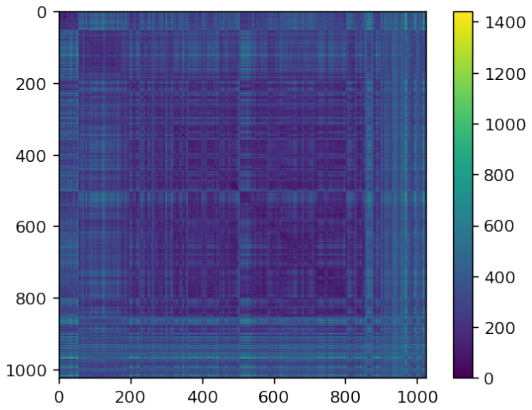

Figura 1. EMD entre imágenes calculadas en claro

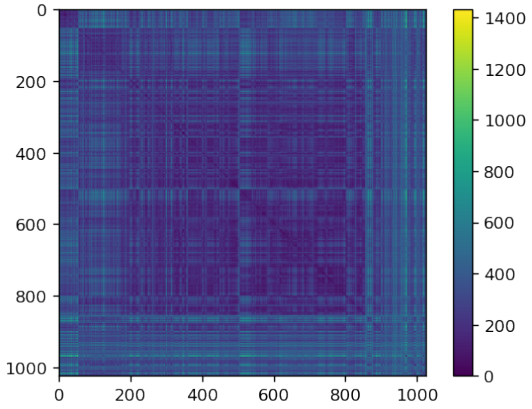

Figura 2. EMD entre imágenes calculadas con nuestro método

Todas las búsquedas de imágenes que se encuentran en la base de datos devolvieron un resultado correcto, aunque algunas búsquedas devolvieron más de un resultado. En la figura 3 se muestra uno de estos ejemplos. Esto se ha producido en 12 casos, un $1,17\,\%$.

De las 8120 imagenes *no* incluidas en la base de datos, 9 devolvieron un resultado. La figura 4 muestra uno de estos ejemplos.

V. Conclusiones

En este trabajo hemos propuesto un esquema de cálculo seguro biparte de la distancia de Wasserstein, más conocida com *Earth Mover's Distance*, entre distribuciones discretas unidimensionales. Para ello, hemos demostrado que el cálculo de la EMD en estos casos se puede reducir al cálculo de tamaño de la intersección de los conjuntos que codifican dichas distribuciones. A partir de este resultado, proponemos usar esquemas existentes para el cálculo seguro de la cardinalidad de la intersección de conjuntos (PSI-CA) para obtener la EMD entre dos distribuciones. Demostramos nuestro esquema con

Figura 3. Falso positivo: para una búsqueda de la imagen de la izquierda, se han devuelto tanto la imagen correcta como la imagen de la derecha.

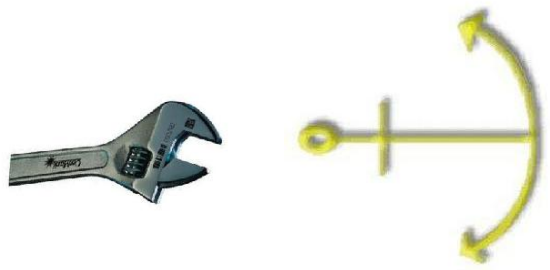

Figura 4. Falso positivo: para una búsqueda de la imagen de la izquierda, se ha devuelto la imagen de la derecha.

un sistema de cifrado buscable para búsquedas inversas de imágenes.

Agradecimientos

Agradecemos las subvenciones de los organismos siguientes: Comisión Europea (project H2020-871042 "SoBigData++"), Generalitat de Catalunya (Premio ICREA Acadèmia al segundo autor y ayuda 2017 SGR 705) y Gobierno de España (proyectos RTI2018-095094-B-C21 y TIN2016-80250-R). Los autores pertenecen a la Cátedra UNESCO de Privacidad de datos, pero las opiniones en este artículo son suyas y no son necesariamente compartidas por UNESCO o los organismos financiadores.

Referencias

[1] Y. Rubner, C. Tomasi, and L. J. Guibas, "A metric for distributions with applications to image databases," in *6th International Conference on Computer Vision*. IEEE, 1998, pp. 59–66.

[2] ——, "The earth mover's distance as a metric for image retrieval," *International Journal of Computer Vision*, vol. 40, no. 2, pp. 99–121, 2000.

[3] X. Wan and Y. Peng, "The earth mover's distance as a semantic measure for document similarity," in *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, 2005, pp. 301–302.

[4] M. Kusner, Y. Sun, N. Kolkin, and K. Weinberger, "From word embeddings to document distances," in *International Conference on Machine Learning*, 2015, pp. 957–966.

[5] N. Li, T. Li, and S. Venkatasubramanian, "t-closeness: Privacy beyond k-anonymity and l-diversity," in *2007 IEEE 23rd International Conference on Data Engineering*. IEEE, 2007, pp. 106–115.

[6] P. Samarati, and L. Sweeney, *Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression,* Research Report, SRI International, 1998.

[7] M. J. Freedman, K. Nissim, and B. Pinkas, "Efficient private matching and set intersection," in *Advances in Cryptology (EUROCRYPT 2004)*. Springer, 2004, pp. 1–19.

[8] A. Blanco, J. Domingo-Ferrer, O. Farras, and D. Sánchez, "Distance computation between two private preference functions," in *IFIP International Information Security Conference*. Springer, 2014, pp. 460–470.

[9] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Advances in Cryptology (EUROCRYPT 1999)*. Springer, 1999, pp. 223–238.

[10] M. J. Freedman, C. Hazay, K. Nissim, and B. Pinkas, "Efficient set intersection with simulation-based security," *Journal of Cryptology*, vol. 29, no. 1, pp. 115–155, 2016.

[11] C. Dong, L. Chen, and Z. Wen, "When private set intersection meets big data: an efficient and scalable protocol," in *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security (CCS'13)*, 2013, pp. 789–800.

[12] B. Pinkas, T. Schneider, and M. Zohner, "Faster private set intersection based on {OT} extension," in *23rd {USENIX} Security Symposium ({USENIX} Security 14)*, 2014, pp. 797–812.

[13] B. H. Bloom, "Space/time trade-offs in hash coding with allowable errors," *Communications of the ACM*, vol. 13, no. 7, pp. 422–426, 1970.

[14] F. Kerschbaum, "Outsourced private set intersection using homomorphic encryption," in *Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security (ASIACCS'12)*, 2012, pp. 85–86.

[15] Ú. Erlingsson, V. Pihur, and A. Korolova, "Rappor: Randomized aggregatable privacy-preserving ordinal response," in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security (CCS'14)*, 2014, pp. 1054–1067.

[16] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," in *2004 Conference on Computer Vision and Pattern Recognition Workshop*. IEEE, 2004, pp. 178–178.