(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 452 735**

(51) Int. CI.:

G06K 9/62 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Fecha de presentación y número de la solicitud europea: **25.08.2006 E 06119599 (6)**

(97) Fecha y número de publicación de la concesión europea: **12.03.2014 EP 1903479**

(54) Título: **Método y sistema para la clasificación de datos utilizando un mapa auto-organizativo**

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
02.04.2014

(73) Titular/es:

MOTOROLA MOBILITY LLC (100.0%)
600 North US Highway 45
Libertyville, IL 60048 , US

(72) Inventor/es:

DARA, ROZITA A.;
KHAN, MOHAMMAD TAUSEEF;
AZIM, JAWAD;
CICCHELLO, ORLANDO y
CORT, GARY P.

(74) Agente/Representante:

DE ELZABURU MÁRQUEZ, Alberto

DESCRIPCIÓN

Método y sistema para la clasificación de datos utilizando un mapa auto-organizativo

La presente descripción se refiere en general a métodos y sistemas para la clasificación de datos utilizando un mapa auto-organizativo. En particular, algunas realizaciones utilizan un mapa auto-organizativo para etiquetar al menos algunos datos no etiquetados dentro de un conjunto de datos.

Se ha mostrado que los algoritmos que aprenden de máquinas resultan ser métodos prácticos para problemas de reconocimiento del mundo real. Se ha probado también que son eficientes en dominios que son altamente dinámicos con respecto a muchos valores y condiciones. Algunos algoritmos que aprenden de máquinas son adecuados para clasificación (o modelización predictiva), mientras que otros han sido desarrollados para propósitos de agrupamiento (o modelización descriptiva). El agrupamiento se utiliza para generar una visión global de la relación de los registros de datos. La salida de tales algoritmos pueden ser varios grupos, donde cada grupo contiene un conjunto de registros homogéneos. Tal como se aplica a la gestión de relación de abonado (CRM Customer Relationship Management, en inglés) analítica, por ejemplo, los grupos pueden comprender grupos de registros de abonado con características similares. Para agrupamiento, no se necesita ningún dato etiquetado. En clasificación, por otro lado, se necesita un conjunto de categorías conocidas, fijas y un grupo de registros etiquetados (conocidos como datos de entrenamiento) para construir un modelo de clasificación. Los modelos de clasificación pueden ser ampliamente utilizados en los sistemas de CRM analíticos para organizar en categorías los registros de usuario en clases predefinidas.

Uno de los obstáculos para la clasificación es la falta de datos etiquetados disponibles. Un problema que aparece en varios dominios de aplicación es la disponibilidad de grandes cantidades de datos no etiquetados en comparación con los relativamente escasos datos etiquetados. Recientemente, se ha propuesto un aprendizaje semi-supervisado con la promesa de resolver este problema y de acelerar la capacidad de los algoritmos de aprendizaje. El aprendizaje semi-supervisado utiliza datos tanto etiquetados como no etiquetados y puede ser aplicado para mejorar el rendimiento del algoritmo de clasificación y de agrupamiento.

Los datos no etiquetados pueden ser recogidos mediante un medio automatizado de varias bases de datos, mientras que los datos etiquetados pueden requerir la introducción de expertos humanos u otros recursos de categorización limitados o costosos. El hecho de que los datos no etiquetados estén fácilmente disponibles, o no sean costosos de recoger, puede resultar atractivo y puede ser deseable el utilizarlos. No obstante, a pesar del natural atractivo de utilizar datos no etiquetados, no es obvio cómo pueden los registros sin etiquetas ayudar a desarrollar un sistema para el propósito de predecir las etiquetas.

Se presta atención a un artículo titulado "A SOM/MLP Hybrid Network that uses Unlabeled Data to Improve Classification Performance", por Stacey et al publicado en Smart Engineering System Design: Neural Networks, Fuzzy Logic, Evolutionary Programming, data Mining and Complex Systems Proceedings of the Artificial Neural Networks in Engineering Conference, XX, XX, vol 10, 5 Noviembre de 2000 páginas 179-184, XP008073219. Este documento describe un planteamiento para utilizar datos no etiquetados para ayudar en el entrenamiento de una red neural supervisada, que implica el uso de un mapa auto-organizativo (SOM Self Organizing Map, en inglés). Los datos son asignados a nodos utilizando el SOM y donde todos los datos etiquetados asignados a un grupo tienen la misma etiqueta, y entonces a los datos no etiquetados asignados al mismo grupo se les da la misma etiqueta. En el caso de nodos ambiguos, los nodos vecinos pueden ser consultados.

DESCRIPCIÓN DE REALIZACIONES PREFERIDAS

Las realizaciones descritas en esta memoria se refieren en general a sistemas y métodos para generar datos de entrenamiento utilizando un mapa auto-organizativo. Los datos de entrenamiento pueden entonces ser utilizados para entrenar a un clasificador para la clasificación de datos. Los datos utilizados para rellenar el mapa auto-organizativo consisten en una pequeña cantidad de datos etiquetados y en una relativamente mucho mayor cantidad de datos no etiquetados. Mediante proximidad de los datos no etiquetados a los datos etiquetados en nodos del mapa auto-organizativo, pueden asignarse etiquetas a los datos no etiquetados. Las realizaciones implementan un modelo de red neural híbrido para combinar un gran conjunto de datos no etiquetados con un pequeño conjunto de registros etiquetados para su uso en clasificación. Pueden aplicarse realizaciones a CRM analítica o en sistemas que rastrean grandes cantidades de datos de usuario, por ejemplo para funciones de auto-rellenado o auto-selección.

Pueden aplicarse también realizaciones a varios usos que pueden ser categorizados como aplicaciones de predicción, regresión o modelización. Pueden utilizarse realizaciones en campos en medicina, previsión (por ejemplo, negocios o el tiempo), ingeniería de software (por ejemplo predicción de defectos de software o modelización de fiabilidad de software), fabricación (por ejemplo optimización y resolución de problemas) y extracción de datos. Pueden también utilizarse realizaciones en áreas financieras, tal como para calificación crediticia y detección de fraude. Pueden también utilizarse realizaciones en campos de bioinformática, tales como análisis de alineamiento de estructura proteica, estudios de genoma y análisis de micro-matriz.

Algunos usos específicos de realizaciones en el campo de la medicina pueden incluir: localización de características comunes relacionadas con la salud en grandes cantidades de datos; previsión mejorada de resultados sobre la base de los datos existentes, tal como tiempo de recuperación de un paciente o cambios en los ajustes de un dispositivo; predicción de la progresión probable de datos médicos a lo largo del tiempo, tal como el crecimiento de una célula o la dispersión de una enfermedad; identificación de características específicas en imágenes médicas, tal como detección de características de ultrasonidos o de rayos-X; y agrupamiento de datos médicos sobre la base de características claves, tales como condiciones demográficas y pre-existentes.

Ciertas realizaciones pueden referirse a un método de etiquetar datos para entrenamiento de un clasificador, que comprende obtención de datos, comprendiendo los datos datos etiquetados y datos no etiquetados; generar un mapa auto-organizativo de los datos; y etiquetar al menos algunos de los datos no etiquetados sobre la base de la proximidad de los datos no etiquetados a datos etiquetados dentro del mapa auto-organizativo para generar datos auto-etiquetados; donde el etiquetado comprende etiquetar datos no etiquetados asociados con cada uno de una pluralidad de nodos en el mapa auto-organizativo con una etiqueta de datos etiquetados asociados con el nodo respectivo; donde el etiquetado comprende también, para cada vecindad alrededor de un nodo asociado con datos etiquetados, determinar si los datos asociados con nodos dentro de una profundidad de vecindad predeterminada tienen diferentes etiquetas; donde el etiquetado comprende también, donde se determina que los nodos dentro de una misma vecindad de profundidad uno están asociados con diferentes datos etiquetados, no etiquetando datos no etiquetados asociados con nodos que son adyacentes a cualquiera de los nodos dentro de la misma vecindad de profundidad uno que se ha determinado que están asociados con los datos etiquetados; donde el etiquetado comprende también, si no se determina que los nodos dentro de una misma vecindad de profundidad uno están asociados con diferentes datos etiquetados, entonces donde se determina que los nodos dentro de una misma vecindad que no es de profundidad uno están asociados con diferentes datos etiquetados, etiquetar datos no etiquetados asociados con nodos que son adyacentes a sólo uno de los nodos dentro de la misma vecindad que no es de profundidad uno que se determina que están asociados con datos etiquetados, de manera que a los datos no etiquetados se les asigna la etiqueta de los datos etiquetados asociados con el un nodo adyacente. El método puede también comprender entrenar a un clasificador basándose en los datos etiquetados y auto etiquetados.

El etiquetado puede estar basado en una relación de proximidad de datos no etiquetados y etiquetados dentro de una vecindad de nodos del mapa auto-organizativo. La cantidad de datos etiquetados puede ser incrementada añadiendo los datos auto-etiquetados y el etiquetado se repite. Por ejemplo, generación y etiquetado pueden ser repetidos un número de veces predeterminado. La generación y/o el etiquetado pueden repetirse hasta que se satisface una condición de terminación predeterminada.

El etiquetado puede ser repetido hasta que se satisface una condición de terminación predeterminada. El método puede también comprender el auto-rellenado de un campo de datos o una selección de usuario utilizando el clasificador. El etiquetado puede también comprender, para cada vecindad alrededor de un nodo asociado con datos etiquetados, si se determina que los datos asociados con nodos dentro de una profundidad de vecindad predeterminada no tienen diferentes etiquetas, etiquetar todos los datos no etiquetados asociados con nodos en la respectiva vecindad con la etiqueta de los datos etiquetados en esa vecindad. La profundidad de vecindad predeterminada puede ser dos, por ejemplo.

Otra realización puede referirse a un sistema para etiquetar datos para su uso en la clasificación de datos, que comprende: al menos un procesador; una memoria accesible para el al menos un procesador y que almacena instrucciones de programa ejecutables por el al menos un procesador, estando las instrucciones de programa dispuestas en una pluralidad de módulos que comprenden un módulo de agrupamiento y un módulo de auto-etiquetado; donde el módulo de agrupamiento está configurado para recibir datos que comprenden datos etiquetados y datos no etiquetados y para generar un mapa auto-organizativo de los datos; y donde el módulo de auto-etiquetado está configurado para etiquetar al menos algunos de los datos no etiquetados sobre la base de la proximidad de los datos no etiquetados a datos etiquetados dentro del mapa auto-organizativo para generar con ello los datos auto-etiquetados; donde el módulo de auto-etiquetado está también configurado para etiquetar datos no etiquetados asociados con cada uno de la pluralidad de nodos en el mapa auto-organizativo con una etiqueta de datos etiquetados asociados con el nodo respectivo; donde el módulo de auto-etiquetado está también configurado, para cada vecindad alrededor de un nodo asociado con datos etiquetados, para determinar si los datos asociados con nodos dentro de una profundidad de vecindad predeterminada tienen diferentes etiquetas; donde el módulo de auto-etiquetado está también configurado, donde se determina que los nodos dentro de una misma vecindad de profundidad uno están asociados con diferentes datos etiquetados, para no etiquetar datos no etiquetados asociados con nodos que son adyacentes a cualquiera de los nodos dentro de la misma vecindad de profundidad uno que se ha determinado que están asociados con los datos etiquetados; y donde el módulo de auto-etiquetado está también configurado para, si los nodos dentro de una misma vecindad de profundidad uno no se determina que están asociados con diferentes datos etiquetados, y donde se determina que los nodos dentro de una misma vecindad que no es de profundidad uno están asociados con diferentes datos etiquetados, etiquetar los datos no etiquetados asociados con nodos que son adyacentes a sólo uno de los nodos dentro de la misma vecindad que no es de profundidad uno que se determina que están asociados con datos etiquetados, de manera que a los datos no etiquetados se les asigna la etiqueta de los datos etiquetados asociados con el un nodo adyacente. El sistema puede también comprender un módulo de clasificación configurado para entrenar a un clasificador basándose en los datos etiquetados y auto-etiquetados.

El etiquetado mediante el módulo de auto-etiquetado puede estar basado en una relación de proximidad de datos no etiquetados y etiquetados dentro de una vecindad de nodos del mapa auto-organizativo. El módulo de auto-etiquetado puede ser configurado para añadir los datos auto-etiquetados a los datos etiquetados y para generar también datos auto-etiquetados a partir de datos no etiquetados. El módulo de auto-etiquetado puede también ser configurado para añadir iterativamente los datos auto-etiquetados a los datos etiquetados y generar otros datos auto-etiquetados hasta que se satisface una condición de terminación predeterminada.

El módulo de agrupamiento puede ser configurado para regenerar el mapa auto-organizativo basándose en los datos etiquetados, los datos no etiquetados y los datos auto-etiquetados. El clasificador puede ser también configurado para auto-completar un campo de datos o una selección de usuario basándose en los datos etiquetados y auto-etiquetados. El módulo de auto-etiquetado puede ser también configurado para asignar una clase a los datos auto-etiquetados sobre la base de los datos etiquetados en la misma vecindad.

El módulo de auto-etiquetado puede ser también configurado para, para cada vecindad alrededor de un nodo asociado con datos etiquetados, si se determina que los datos asociados con nodos dentro de una profundidad de vecindad predeterminada no tienen diferentes etiquetas, etiquetar todos los datos no etiquetados asociados con nodos en la respectiva vecindad con la etiqueta de los datos etiquetados en esa vecindad. La profundidad de vecindad predeterminada puede ser dos.

Ciertas realizaciones pueden también referirse a almacenes legibles por ordenador que almacenan instrucciones de programa de ordenador las cuales, cuando son ejecutadas por al menos un procesador, hacen que el al menos un procesador ejecute un método que comprende: obtener datos, comprendiendo los datos datos etiquetados y datos no etiquetados; generar un mapa auto-organizativo de los datos; etiquetar al menos algunos de los datos no etiquetados sobre la base de la proximidad de los datos no etiquetados a datos etiquetados dentro del mapa auto-organizativo para generar datos auto-etiquetados; El método puede también comprender entrenar a un clasificador basándose en los datos etiquetados y auto-etiquetados.

En algunos casos, los datos necesitan ser preprocesados antes de construir un modelo de clasificación y de analizar los resultados. Dependiendo del tipo de datos, pueden requerirse diferentes tipos de preprocesamiento. Si los datos de fuente son puramente numéricos, puede ser necesaria solamente normalización y selección de características. Si los campos de datos de interés en la base de datos no son numéricos, entonces la tarea de preprocesamiento de datos puede ser más complicada. Pueden emplearse herramientas de preprocesamiento automatizadas para transformar los datos en un formato numérico adecuado. Técnicas de preprocesamiento de datos adecuadas para su uso en clasificación y/o agrupamiento serán comprendidas por los expertos en la materia.

Una técnica de preprocesamiento de texto automatizado adecuada puede emplear tokenización (o rotura de un texto en tokens, o componentes léxicos), eliminación de palabra reservada, derivación, comprobación de deletreo y construcción de registros de características. Puede utilizarse un método de TFxIDF (frecuencia de término multiplicado por la frecuencia de documento inversa) para el cálculo de la frecuencia de término y la ganancia de información puede ser utilizada para selección de característica.

En las realizaciones descritas, puede utilizarse una forma de mapa auto-organizativo (SOM - Self-Organizing Map, en inglés) de red neural para inferir una clase asociada con datos no etiquetados sobre la base de su proximidad a datos etiquetados dentro del SOM. Un SOM es una red de nodos en forma de hoja que a su vez está sintonizada con varios registros de datos de entrenamiento a través de un proceso de aprendizaje. En la generación (también llamada entrenamiento) del SOM, uno o más registros, tanto etiquetados como no etiquetados, pueden ser asociados con un nodo del SOM.

El SOM puede por consiguiente organizar los registros de entrenamiento en grupos, en los cuales similares registros están situados en nodos cercanos entre sí. Esta única característica de un SOM puede ser utilizada, junto con el pequeño conjunto de datos etiquetados, para obtener grupos más precisos. Además, los grupos resultantes y un pequeño conjunto de datos etiquetados pueden ser utilizados para clasificar registros no etiquetados y obtener registros etiquetados adicionales (registros auto-etiquetados). Los datos auto-etiquetados se utilizan para reformular los grupos o para proporcionar un conjunto de datos mayor para una tarea de clasificación.

El SOM puede definir de manera efectiva una relación entre nodos espacialmente adyacentes que puede ser aprovechada para el etiquetado. En las realizaciones descritas, se asignan etiquetas a registros no etiquetados en un nodo examinando los registros etiquetados en el propio nodo y en nodos vecinos.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

A continuación de describen realizaciones con más detalle, a modo de ejemplo, con referencia a los dibujos que se acompañan, en los cuales:

La FIG. 1 es un diagrama de bloques de un sistema para su uso en generar datos de entrenamiento para clasificación;

la FIG. 2 es un diagrama de flujo de un método de generar datos de entrenamiento para clasificación;

la FIG. 3 es un diagrama de flujo de un método de auto-etiquetado de datos no etiquetados;

la FIG. 4 es una ilustración esquemática de la generación de un mapa auto-organizativo utilizando datos etiquetados y no etiquetados;

la FIG. 5 es una ilustración esquemática del auto-etiquetado de registros de datos utilizando un mapa auto-organizativo;

la FIG. 6 es una ilustración esquemática del re-entrenamiento de un mapa auto-organizativo utilizando datos etiquetados y auto-etiquetados combinados, así como datos no etiquetados;

la FIG. 7 es una ilustración esquemática de un ejemplo de un mapa auto-organizativo que utiliza una topología de vecindad rectangular;

la FIG. 8 es una ilustración esquemática de un mapa auto-organizativo de ejemplo que utiliza una topología de vecindad rectangular, en la cual se asignan etiquetas a registros no etiquetados en nodos dentro de una profundidad de vecindad de uno alrededor de un nodo etiquetado;

la FIG. 9 es una ilustración esquemática de un mapa auto-organizativo de ejemplo que utiliza una topología de vecindad rectangular y que muestra un escenario en el que los nodos de vecindad tienen diferentes etiquetas;

la FIG. 10 es una ilustración esquemática de un mapa auto-organizativo de ejemplo que utiliza una topología de vecindad rectangular y que muestra un escenario en el que los nodos etiquetados no son directamente vecinos entre sí; y

la FIG. 11 es un diagrama de flujo de un método de generar un conjunto inicial de datos etiquetados.

En referencia ahora a la FIG. 1, se muestra un diagrama de bloques de un sistema 100 para su uso en la clasificación de datos. El sistema 100 comprende un sistema de servidor 110, una base de datos 120 accesible para el sistema de servidor 110 y una interfaz de usuario 190 en comunicación con el sistema de servidor 110. La base de datos 120 almacena una cantidad de datos de entrada 125. La base de datos 120 puede ser distribuida o discreta. Los datos de entrada 125 pueden incluir registros de datos tanto etiquetados como no etiquetados. Inicialmente, los datos de entrada 125 pueden ser no etiquetados pero, a continuación de la generación de un conjunto de datos etiquetados inicial y la generación de datos auto-etiquetados (que se describe con más detalle a continuación), cada vez más datos no etiquetados pueden convertirse en etiquetados.

El sistema de servidor 110 comprende uno o más procesadores 130 y una memoria 140. El sistema de servidor 110 puede ser un sistema de procesamiento distribuido o virtual. Alternativamente, el sistema de servidor 110 puede corresponder a un sistema de ordenador discreto, tal como un ordenador personal o un dispositivo electrónico móvil. Los procesadores 130 pueden ser operados en paralelo y pueden ser distribuidos, por ejemplo. Alternativamente, sólo un procesador 130 puede ser empleado, por ejemplo para realizaciones implementadas mediante un único dispositivo de cálculo, tal como un ordenador personal o un dispositivo electrónico móvil.

La memoria 140 es un almacén no volátil que comprende instrucciones de programa almacenadas las cuales, cuando son ejecutadas por los procesadores 130, hacen que los procesadores 130 realicen varias funciones, como se describe a continuación. Al menos algunas de las instrucciones de programa almacenadas en la memoria 140 están organizadas en módulos de software. Tales módulos incluyen un módulo de preprocesamiento 150, un módulo de agrupamiento 160, un módulo de auto-etiquetado 170 y un módulo de clasificación 180.

La operación del sistema 100 se describe con más detalle a continuación, en referencia también a la FIG. 2, que es un diagrama de flujo de un método 200 de generar datos de entrenamiento para su uso en clasificación.

En la etapa 205, el módulo de preprocesamiento 150 extrae los datos de entrada 125 de la base de datos 120 y procesa los datos en la etapa 215 de acuerdo con las técnicas de preprocesamiento existentes, como se ha descrito anteriormente, si se requiere, en la etapa 210. El preprocesamiento de los datos de entrada 125 es llevado a cabo sólo si es requerido por el formato y/o el contenido de los datos de entrada 125. Por ejemplo, si los datos de entrada 125 contienen registros que tienen un campo de texto libre, el preprocesamiento del pre-texto necesitará ser realizado por el módulo de preprocesamiento 150. Los datos de entrada 125 pueden ser selectivamente extraídos mediante una pregunta de la base de datos 120 ó mediante una descarga no selectiva de la base de datos 120. La pregunta puede ser introducida a través de la interfaz de usuario 190, por ejemplo.

Una vez que se realiza cualquier preprocesamiento necesario en la etapa 215, el módulo de preprocesamiento 150 proporciona datos de un formato predefinido al módulo de agrupamiento 160. El módulo de agrupamiento 160 toma los datos de entrada 125 como procesados por el módulo de preprocesamiento 150 y genera un conjunto de datos etiquetados inicial en la etapa 220 a partir de un relativamente pequeño subconjunto de datos de entrada 125. La generación del conjunto de datos etiquetados inicial se describe con más detalle a continuación, con referencia a la FIG. 11. Como alternativa a la generación de los datos etiquetados iniciales, puede ser obtenido a partir de un almacén de datos pre-existentes o puede ser proporcionado por un experto del dominio, por ejemplo.

Una vez que el conjunto de datos etiquetados inicial es generado (si no se obtiene de otra forma), el módulo de almacenamiento 160 genera un mapa auto-organizativo utilizando los datos etiquetados y no etiquetados en la etapa 225. Los métodos de generación (también llamada entrenamiento) de un mapa auto-organizativo utilizando un conjunto de datos resultarán evidentes para los expertos en la materia. La generación de un mapa auto-organizativo se muestra esquemáticamente en la FIG. 4, en la cual se utilizan datos etiquetados 410 y datos no etiquetados 420 como datos de entrada 430 para entrenar al SOM 440, con el resultado de que los datos etiquetados 410 y los datos no etiquetados 420 son asignados a uno o más nodos 445 en el SOM 440.

Una vez que el mapa auto-organizativo 440 es entrenado con los datos etiquetados 410 y los datos no etiquetados 420 en la etapa 225, el módulo de auto-etiquetado 170 analiza las posiciones de los registros no etiquetados asignadas a nodos en el SOM 440 con respecto a los nodos que tienen registros etiquetados asociados a ellos para generar datos auto-etiquetados en la etapa 230. El método de generación de datos auto-etiquetados en la etapa 230 se describe con más detalle a continuación, con referencia al diagrama de flujo de la FIG. 3, la ilustración esquemática de la FIG. 5 y los ejemplos mostrados en las FIGS. 8 a 10.

Una vez que los datos auto-etiquetados son generados en la etapa 230, el módulo de auto-etiquetado 170 determina en la etapa 235 si se requiere otra generación de datos auto-etiquetados, bien repitiendo la etapa 230 con el mismo mapa auto-organizativo o bien re-entrenando el mapa auto-organizativo en la etapa 225 utilizando los datos recién auto-etiquetados, los datos etiquetados y los datos aún no etiquetados. Re-entrenar al SOM en la etapa 225 puede resultar en la reorganización de algunos de los datos en los nodos del mapa debido a las características de aleatorización normalmente aplicadas durante el entrenamiento de un mapa auto-organizativo. Otros datos auto-etiquetados pueden ser generados a continuación en la etapa 230 sobre la base del SOM re-entrenado en la etapa 225.

En la etapa 235, puede decidirse repetir sólo la etapa 230, en cuyo caso los datos previamente auto-etiquetados son añadidos a los datos etiquetados y utilizados para generar más datos auto-etiquetados en la etapa 230.

La repetición de las etapas 225 a 235 ó 230 a 235 pueden denominarse re-etiquetado. Las iteraciones de re-etiquetado pueden ser extendidas hasta profundidades arbitrarias. No obstante, existirá normalmente un punto en el cual no se pueden asignar etiquetas al restante conjunto de datos no etiquetados. A medida que el número de iteraciones de re-etiquetado aumenta, la probabilidad de etiquetas contradictorias en nodos vecinos aumenta también.

Pueden tomarse en consideración las siguientes medidas para evitar en lo posible un etiquetado incorrecto, y obtener resultados más fiables:

- Limitar las iteraciones del re-etiquetado a una cantidad pequeña, fija, por ejemplo, 4.

- Optimizar el preprocesamiento de datos empleando un conocimiento previo para ponderar palabras clave (distinto del método de TFxIDF). Esta optimización puede resultar en clases con mayores distancias inter-clases.

- Si dos nodos vecinos son asignados a diferentes clases, pueden no ser utilizados para el proceso de etiquetado.

- El SOM puede ser re-entrenado en cada proceso de re-etiquetado para aprovechar la nueva aleatorización de peso y los cambios en el grupo.

Los criterios de terminación para terminar el re-etiquetado pueden incluir los siguientes: el número de iteraciones de las etapas 225 a 235 ó 230 a 235 ha alcanzado un límite de iteración predeterminado, tal como 3, 4 ó 5, por ejemplo; las iteraciones de re-etiquetado han alcanzado un punto en el cual no puede asignarse ninguna etiqueta nueva al conjunto restante de datos no etiquetados; o se ha obtenido un número predeterminado de registros etiquetados. Este último criterio es relevante cuando la cantidad de los datos de entrenamiento es conocida de antemano.

Una vez que el módulo de auto-etiquetado 170 determina que existe un criterio de terminación para terminar el re-etiquetado, en la etapa 235, los datos etiquetados, que incluyen cualquier dato auto-etiquetado, se pasa al módulo de clasificación 180, que utiliza los datos etiquetados para entrenar a un clasificador. El clasificador puede ser entrenado utilizando un algoritmo de red neural, tal como el perceptrón de multi-capa (MLP - Multi-Layer Perceptron, en inglés), por ejemplo.

Una vez que el módulo de clasificación 180 entrena al clasificador basándose en los datos etiquetados en la etapa 240, el clasificador puede ser utilizado para funciones de predicción con respecto a datos desconocidos, en la etapa 245. Tales funciones de predicción pueden incluir, por ejemplo, entrada de usuario de auto-rellenado, o de auto-selección. El clasificador puede ser también utilizado para propósitos de análisis de datos y de toma de decisiones en las aplicaciones de ejemplo descritas anteriormente. Una vez que el clasificador ha sido entrenado en la etapa 240, el módulo de clasificación 180 actualiza la base de datos 120 con los datos etiquetados para un subsiguiente uso en el agrupamiento y/o la clasificación (incluyendo la predicción).

En referencia ahora a la FIG. 3, se describe con más detalle un método de generación de datos auto-etiquetados (etapa 230). El método se inicia en la etapa 305, en cuyo módulo de auto-etiquetado 170 sitúa todos los registros de datos etiquetados en el SOM 540, como se muestra en la FIG. 5. La FIG. 5 muestra el SOM 540 tras el entrenamiento (o la generación) en la etapa 220. Cada uno de los nodos del SOM 540 puede tener ninguno, uno o más (posiblemente muchos) registros de datos asociados a él (o asignados a él).

La FIG. 5 muestra los datos etiquetados originales entrenados en el SOM 540, con los datos no etiquetados, donde los nodos que contienen registros no etiquetados están indicados por el número de referencia 510. Los datos etiquetados originales están divididos en tres clases separadas con el propósito de ilustración. Los nodos que tienen registros etiquetados correspondientes a la clase 1 están indicados por el número de referencia 501. Los nodos que tienen registros etiquetados correspondientes a la clase 2 están indicados por el número de referencia 502. Los nodos que tienen registros etiquetados correspondientes a la clase 3 están indicados por el número de referencia 503. Algunos nodos tienen datos tanto etiquetados como no etiquetados. Por ejemplo, los nodos 513 contienen registros etiquetados correspondientes a la clase 3, así como registros no etiquetados. De manera similar, el nodo 511 contiene registros etiquetados correspondientes a la clase 1, junto con registros no etiquetados. En estos nodos, a todos los registros no etiquetados se les asignará la etiqueta del registro etiquetado en el nodo en la etapa 310, como se describe en lo que sigue. Algunos nodos del SOM 540 pueden no tener ningún registro asociado a ellos, y están indicados por el número de referencia 506.

Dependiendo de la proximidad de los nodos que tienen registros etiquetados a nodos que tienen registros no etiquetados, a los registros no etiquetados puede serles asignada una etiqueta de acuerdo con el método que se describe en lo que sigue. Tal asignación de etiqueta puede corresponder a la asignación de una clase a los registros no etiquetados. Tales registros no etiquetados se convierten entonces en datos auto-etiquetados 520, que pueden ser divididos en un número de clases, tales como las tres clases mostradas en las FIGS. 4 a 6.

Una vez que los registros etiquetados son situados en el SOM 540 en la etapa 305, se asignan etiquetas a cada uno de los registros en cada nodo que tiene un registro etiquetado asociado con él, en la etapa 310. Así, si un nodo tiene 99 registros no etiquetados y un registro etiquetado, a todos los 99 registros sin etiquetar se les asignará la etiqueta del un registro etiquetado.

En la etapa 315, para cada nodo que tiene un registro etiquetado, el módulo de auto-etiquetado 170 sitúa los nodos vecinos en el SOM 540 de acuerdo con una topología de vecindad especificada. La topología de vecindad puede ser rectangular o hexagonal, por ejemplo, y puede tener una profundidad de vecindad especificada. La profundidad de vecindad puede ser uno o dos, por ejemplo, y especifica el número de nodos adyacentes lejos del nodo bajo consideración que extiende la vecindad. Por ejemplo, para una topología de vecindad rectangular con una profundidad de vecindad de uno, la vecindad consistiría en ocho nodos de vecindad alrededor del nodo bajo consideración. Para una profundidad de vecindad de dos, la vecindad consistiría en los ocho nodos de vecindad de profundidad uno y otros dieciséis nodos en la profundidad dos dispuestos rectangularmente alrededor de los nodos de profundidad uno. La FIG. 7 ilustra un SOM 700 de ejemplo que tiene una topología rectangular, donde los nodos en la profundidad de vecindad de uno están conectados mediante una línea negra en forma de un rectángulo.

En la siguiente etapa 315, el módulo de auto-etiquetado 170 determina, para un nodo particular que tiene registros no etiquetados, si dos registros con diferentes etiquetas están situados en la misma vecindad. Si no, en la etapa 325, el módulo de auto-etiquetado 170 asigna la misma etiqueta que el registro etiquetado a los registros no etiquetados situados en (asociados con) los nodos vecinos, generando por ello registros auto-etiquetados. Debe observarse que, hasta el punto en el que la etapa 310 implica asignar etiquetas a registros no etiquetados que comparten un nodo con un registro etiquetado, esto resulta también en el auto-etiquetado de registros.

La FIG. 8 ilustra un SOM 800 de ejemplo en el cual se aplica la etapa 325 para etiquetar registros no etiquetados asociados con el nodo vecino de nodos (2, 2) en una profundidad de vecindad de uno. A cada uno de los registros no etiquetados asociado con los nodos indicados con una marca de comprobación en la FIG. 8 se les asigna así la misma etiqueta que a los registros etiquetados del nodo (2, 2).

Si, en la etapa 320, el módulo de auto-etiquetado 170 determina que dos registros con diferentes etiquetas están en la misma vecindad, el módulo de auto-etiquetado 170 determina entonces, en la etapa 330, si los registros con diferentes etiquetas están dentro de una profundidad de vecindad de uno. Si es así, entonces en la etapa 335, el módulo de auto-etiquetado 170 determina que las etiquetas no deberían ser asignadas a registros no etiquetados en la vecindad de cada uno de los nodos que tienen registros etiquetados.

La FIG. 9 ilustra un SOM 900 de ejemplo, en el que la etapa 335 es ejecutada. En el SOM 900, los nodos etiquetados (2, 2) y (2, 3) están situados adyacentes entre sí y dentro de una profundidad de vecindad de uno. En este escenario, a ninguno de los registros no etiquetados de los nodos de la profundidad de vecindad uno alrededor de los nodos que tienen registros etiquetados se les asignan etiquetas. Los nodos que tienen registros no etiquetados a los que no se les asignan etiquetas se indican en la FIG. 9 mediante cruces.

Si, en la etapa 330, el módulo de auto-etiquetado 170 determina que los dos registros con diferentes etiquetas en la misma vecindad no están dentro de una profundidad de vecindad de uno con respecto al otro, entonces en la etapa

340, el módulo de auto-etiquetado 170 identifica los nodos de vecindad que tampoco son vecinos de un nodo que tiene unos registros etiquetados y asigna la etiqueta del registro etiquetado a todos los registros no etiquetados en aquellos nodos identificados, generando con ello registros auto-etiquetados.

La FIG. 10 ilustra un SOM 1000 de ejemplo, en el que la etapa 340 es ejecutada. En la FIG. 10, los nodos (2, 2) y (2, 4) tienen registros etiquetados de manera diferente, aunque esos nodos no son inmediatamente adyacentes uno a otro. Por el contrario, esos nodos están dentro de una profundidad de vecindad de dos uno respecto a otro. De acuerdo con esto, los nodos (1, 3), (2, 3) y (3, 3) que son adyacentes a los dos nodos que tienen registros etiquetados no se utilizan para propósitos de auto-etiquetado. Por otro lado, se utilizan registros no etiquetados en nodos que son adyacentes sólo a uno de los nodos etiquetados de manera diferente, pero no a ambos, con el propósito de auto-etiquetado. Los nodos que tienen registros no etiquetados a los que se les asignan etiquetas se indican en la FIG. 10 mediante marcas de comprobación, mientras que aquéllos a los que no se les han asignado etiquetas están indicados mediante cruces.

Una vez que los registros auto-etiquetados se han generado (o no) en las etapas 320 a 340, el módulo de auto-etiquetado 170 comprueba, en la etapa 345, si todos los nodos han sido considerados para propósitos de asignación de etiquetas a registros no etiquetados en nodos vecinos. Si no, las etapas 315 a 340 se repiten, según sea apropiado. Si todos los nodos etiquetados han sido considerados, entonces en la etapa 350, los registros auto-etiquetados son añadidos a los registros etiquetados originales para proporcionar un conjunto de datos etiquetados más grande. Como se ha mencionado en relación con la FIG. 2, la generación de datos auto-etiquetados de acuerdo con la etapa 230 puede ser repetida iterativamente, sola o en combinación con el re-entrenamiento del SOM en la etapa 225.

El re-entrenamiento del SOM se ilustra esquemáticamente en la FIG. 6, en la cual nuevos datos etiquetados 610, que comprenden datos originales 410 y datos auto-etiquetados 520, son combinados con datos todavía no etiquetados 620 como datos de entrada 630 para generar un SOM 640 re-entrenado. Un potencial resultado del re-entrenamiento del SOM es que algunos de los nodos o vecindades pueden ser redefinidos y posiblemente re-etiquetados. El SOM re-entrenado 640 puede entonces ser utilizado para otro auto-etiquetado de los registros no etiquetados, como se ha descrito anteriormente en relación con la FIG. 3. Esto puede resultar en una reasignación de clase para registros etiquetados o auto-etiquetados asociados con algunos nodos.

En referencia ahora a la FIG. 11, se describe con más detalle un método de generación de datos auto-etiquetados (etapa 220). El método se inicia en la etapa 1110, en la cual un pequeño conjunto de datos de entrenamiento 1115 es seleccionado a partir de un conjunto de datos no etiquetados 1105 grande agrupando el módulo 160 a continuación del preprocesamiento de los conjuntos de datos no etiquetados 1105. El módulo de agrupamiento 160 puede seleccionar los datos de entrenamiento 1115 aleatoriamente o mediante agrupamiento o de acuerdo con criterios predeterminados, por ejemplo de manera que se consiga una selección ampliamente representativa de los datos no etiquetados.

En la etapa 1120, el módulo de agrupamiento 160 establece un conjunto de reglas para determinar cómo etiquetar datos no etiquetados. Una regla de ejemplo puede ser como sigue:

Regla 1: Si {Relación instalación es 1 y Relación_Error de usuario es Y y

Queja total es Z} y Nombre-de-producto es Prod 1, el registro pertenece a la etiqueta 1.

Tales reglas pueden ser denominadas reglas difusas puesto que aplican una forma de lógica difusa. Las reglas difusas pueden ser establecidas para satisfacer ciertos objetivos y logros del sistema 100. Las reglas pueden ser extraídas de conjuntos predefinidos de reglas dentro de una librería de reglas y elegidas para adaptarse a un tipo particular de datos de entrada. En la etapa 1125, las reglas se utilizan para asignar etiquetas a los datos de entrenamiento 1115 seleccionados y para generar con ello los datos etiquetados iniciales en la etapa 1130.

Resultará evidente que pueden realizarse variaciones y modificaciones a las realizaciones descritas e ilustradas en esta memoria sin separarse del alcance definido en las reivindicaciones adjuntas.

REIVINDICACIONES

1. Un método (220) implementado en un ordenador, de etiquetar datos para el entrenamiento de un clasificador, que comprende:

obtener datos, comprendiendo los datos datos etiquetados (410) y datos no etiquetados (420);

generar (225) un mapa auto-organizativo de los datos; y

y etiquetar (230) al menos algunos de los datos no etiquetados (510) sobre la base de la proximidad de los datos no etiquetados (510) a los datos etiquetados (501, 502, 503) dentro del mapa auto-organizativo (540) para generar datos auto-etiquetados (520);

donde el etiquetado (230) comprende etiquetar datos no etiquetados asociados con cada uno de una pluralidad de nodos (513, 511) en el mapa auto-organizativo (540) con una etiqueta de datos etiquetados asociados con el nodo respectivo (513, 511);

el etiquetado (230) comprende también, para cada vecindad alrededor de un nodo asociado con datos etiquetados, determinar (320) si los datos asociados con nodos dentro de una profundidad de vecindad predeterminada tienen diferentes etiquetas y, si no, etiquetar (325) todos los datos no etiquetados asociados con nodos en la respectiva vecindad con la etiqueta de los datos etiquetados en esa vecindad.;

donde el etiquetado (230) comprende también, donde se determina que los nodos dentro de una misma vecindad de profundidad uno tienen datos etiquetados de manera diferente, no etiquetar (335) datos no etiquetados dentro de los nodos que están dentro de la citada vecindad de profundidad uno,

y si no, si se determina que los nodos dentro de una misma vecindad de profundidad uno no tienen datos etiquetados de manera diferente, entonces donde se determina que los nodos dentro de una misma vecindad que no es de profundidad uno tienen datos etiquetados de manera diferente, etiquetar (340) datos no etiquetados dentro de nodos que son adyacentes sólo a uno de los nodos dentro de la misma vecindad que no es de profundidad uno que se ha determinado que tiene datos etiquetados, de manera que a los datos no etiquetados se les asigna la etiqueta de los datos etiquetados dentro de un nodo adyacente.

2. El método de la reivindicación 1, que comprende también entrenar a un clasificador basándose en datos etiquetados y auto-etiquetados.

3. El método de la reivindicación 1 ó la reivindicación 2, en el que el etiquetado (230) se basa en una relación de proximidad de datos no etiquetados y etiquetados dentro de una vecindad de nodos del mapa auto-organizativo.

4. El método de la reivindicación 1 ó la reivindicación 2, en el que la cantidad de datos etiquetados se incrementa añadiendo (350) los datos auto-etiquetados a los datos etiquetados (410) y el etiquetado (230) se repite.

5. El método de la reivindicación 4, donde la generación (225) y/o el etiquetado (230) se repite o repiten hasta que se satisface una condición de terminación predeterminada.

6. El método de una cualquiera de las reivindicaciones 1 a 4, donde la generación y/o el etiquetado (230) se repite o repiten hasta que se satisface una condición de terminación predeterminada.

7. El método de una cualquiera de las reivindicaciones 1 a 5, que comprende también auto-rellenar un campo de datos o una selección de usuario utilizando un clasificador.

8. El método de la reivindicación 3, en el que el etiquetado (230) comprende asignar una clase a los datos auto-etiquetados sobre la base de los datos etiquetados en la misma vecindad.

9. El método de cualquiera de las reivindicaciones precedentes, en el que el etiquetado (230) comprende también, para cada vecindad alrededor de un nodo asociado con datos etiquetados, si se determina que los datos asociados con nodos dentro de una profundidad de vecindad predeterminada no tienen etiquetas diferentes, etiquetar (325) todos los datos no etiquetados asociados con nodos en la respectiva vecindad con la etiqueta de los datos etiquetados en esa vecindad.

10. El método de la reivindicación 9, en el que la profundidad de vecindad predeterminada es uno.

11. El método de la reivindicación 9, en el que la profundidad de vecindad predeterminada es dos.

12. El método de una cualquiera de las reivindicaciones 1 a 11, donde los datos etiquetados (410) se generan (220) a partir de los datos no etiquetados (1105) sobre la base de reglas difusas.

13. El método de la reivindicación 12, en el que la generación (220) de los datos etiquetados (410) a partir de datos no etiquetados (1105) comprende:

seleccionar (1110) un conjunto de datos de entrenamiento a partir de datos no etiquetados (1105);

establecer (1120) las reglas difusas como un conjunto de reglas para determinar cómo etiquetar los datos de entrenamiento no etiquetados (1110); y

asignar (1125) etiquetas a los datos de entrenamiento no etiquetados (1110) basándose en las reglas difusas para generar con ello los datos etiquetados (410).

14. Un sistema (100) para etiquetar datos para su uso en la clasificación configurado para llevar a cabo el método de cualquiera de las reivindicaciones 1 a 13.

15. Almacén legible por un ordenador que almacena instrucciones de programa de ordenador (180) las cuales, cuando son ejecutadas por al menos un procesador (130), hacen que el al menos un procesador (130) ejecute el método de una cualquiera de las reivindicaciones 1 a 13.

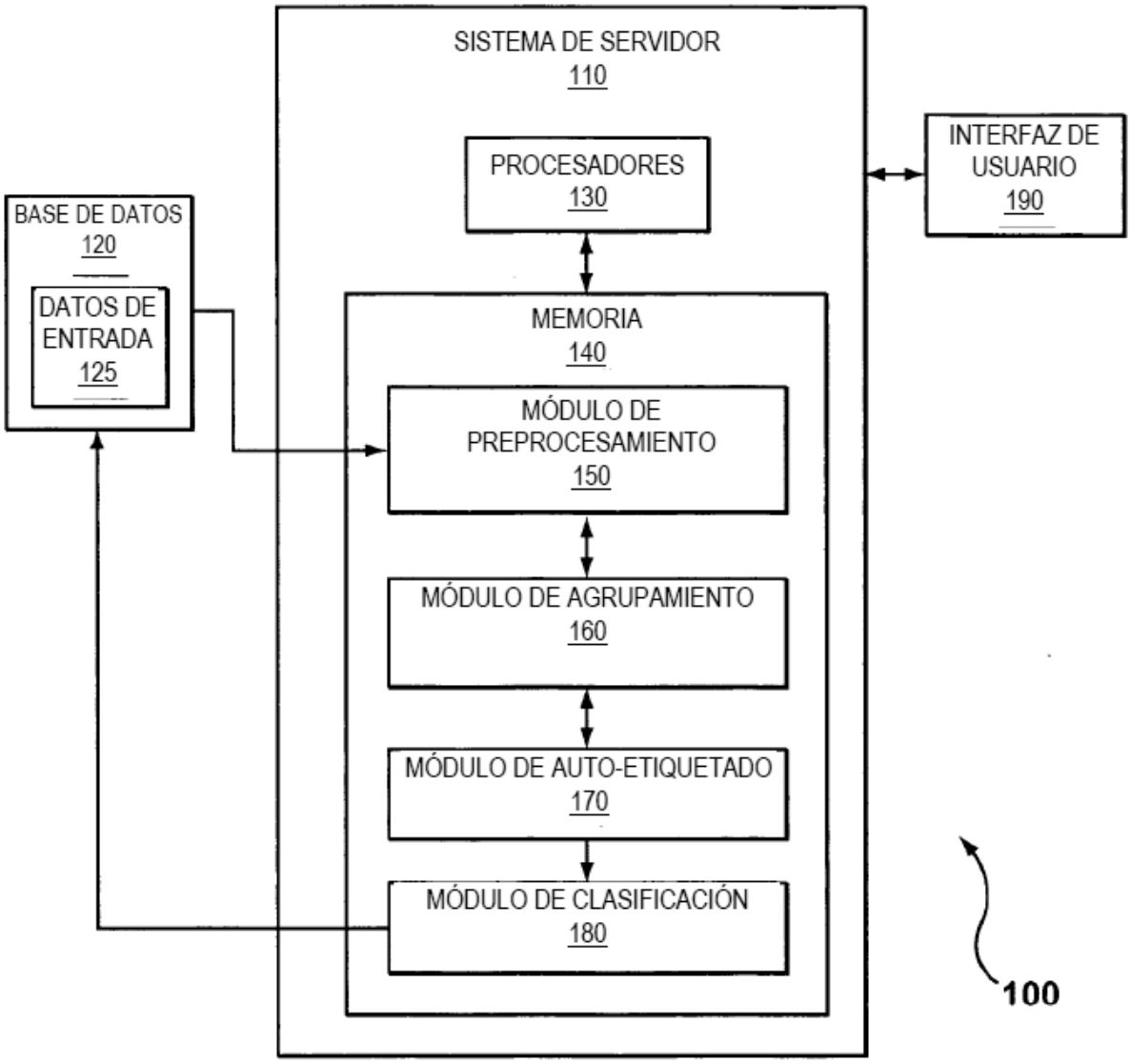
SISTEMA DE SERVIDOR
110
PROCESADORES
130
INTERFAZ DE USUARIO
190
BASE DE DATOS
120
DATOS DE ENTRADA
125
MEMORIA
140
MÓDULO DE PREPROCESAMIENTO
150
MÓDULO DE AGRUPAMIENTO
160
MÓDULO DE AUTO-ETIQUETADO
170
MÓDULO DE CLASIFICACIÓN
180
100

FIG. 1

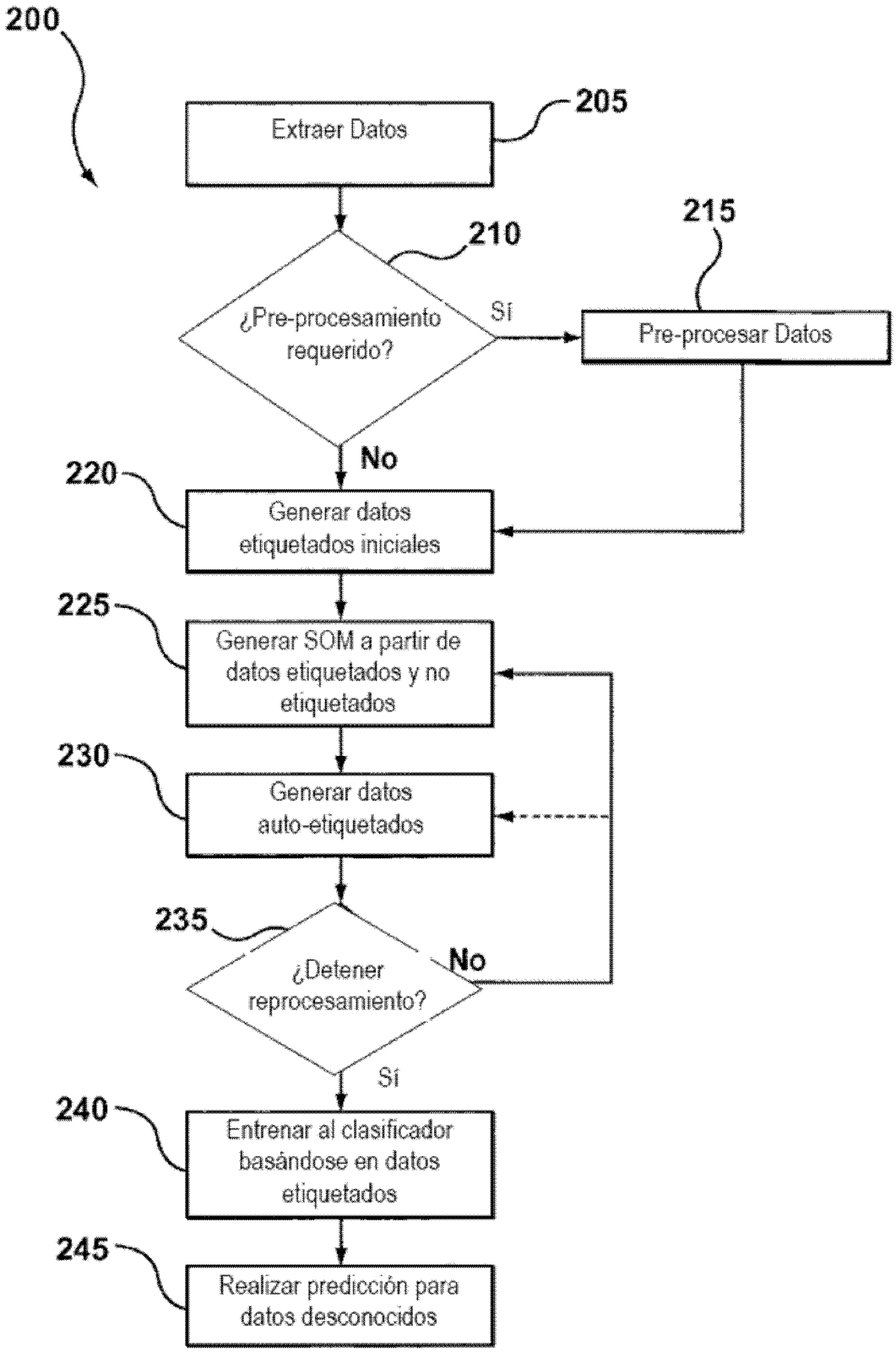
200
Extraer Datos
205
210
¿Pre-procesamiento requerido?
Sí
215
Pre-procesar Datos
No
220
Generar datos etiquetados iniciales
225
Generar SOM a partir de datos etiquetados y no etiquetados
230
Generar datos auto-etiquetados
235
¿Detener reprocesamiento?
No
Sí
240
Entrenar al clasificador basándose en datos etiquetados
245
Realizar predicción para datos desconocidos

FIG. 2

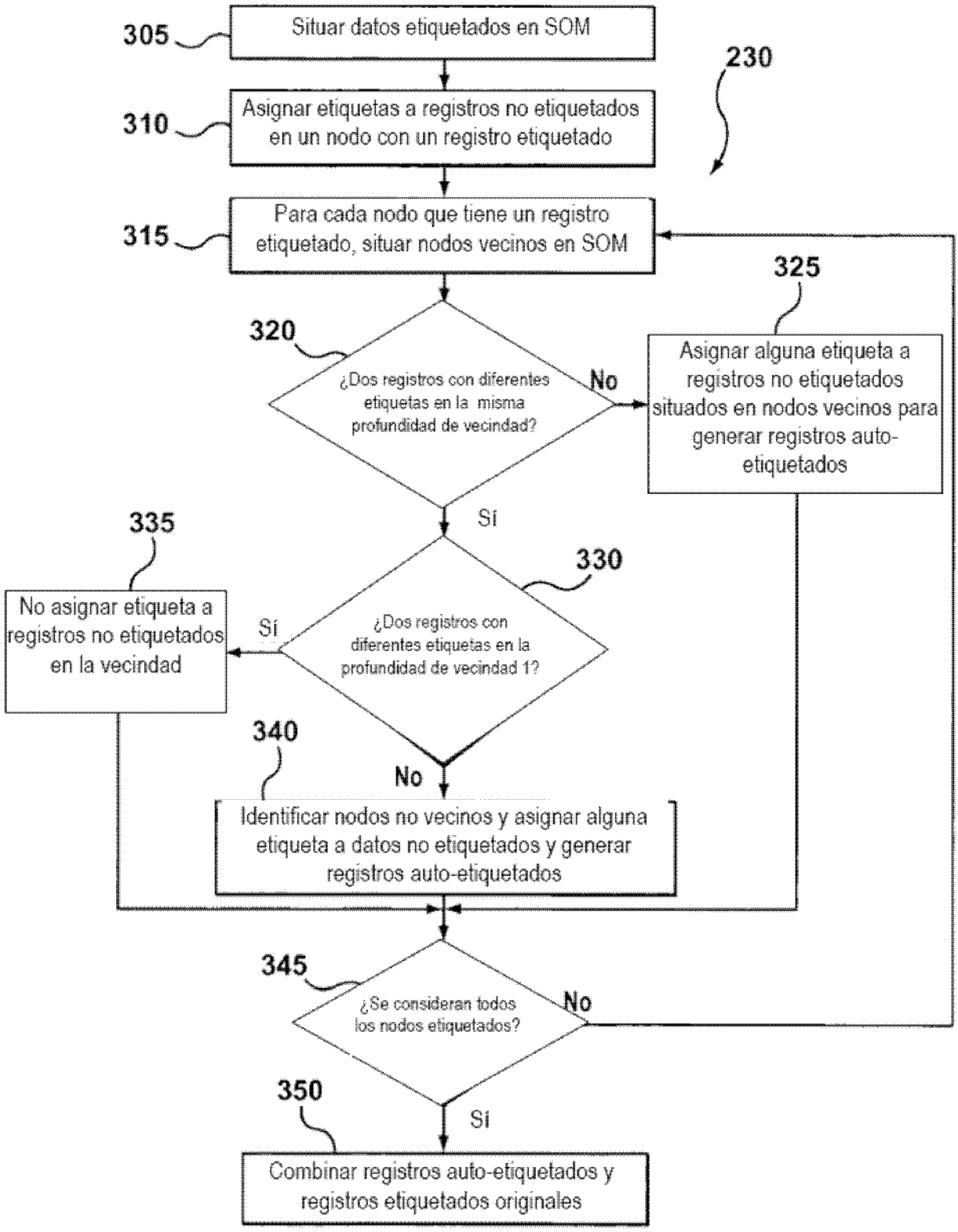
305
Situar datos etiquetados en SOM
230
310
Asignar etiquetas a registros no etiquetados en un nodo con un registro etiquetado
315
Para cada nodo que tiene un registro etiquetado, situar nodos vecinos en SOM
325
320
¿Dos registros con diferentes etiquetas en la misma profundidad de vecindad?
No
Asignar alguna etiqueta a registros no etiquetados situados en nodos vecinos para generar registros auto-etiquetados
Sí
335
330
No asignar etiqueta a registros no etiquetados en la vecindad
Sí
¿Dos registros con diferentes etiquetas en la profundidad de vecindad 1?
340
No
Identificar nodos no vecinos y asignar alguna etiqueta a datos no etiquetados y generar registros auto-etiquetados
345
¿Se consideran todos los nodos etiquetados?
No
350
Sí
Combinar registros auto-etiquetados y registros etiquetados originales

FIG. 3

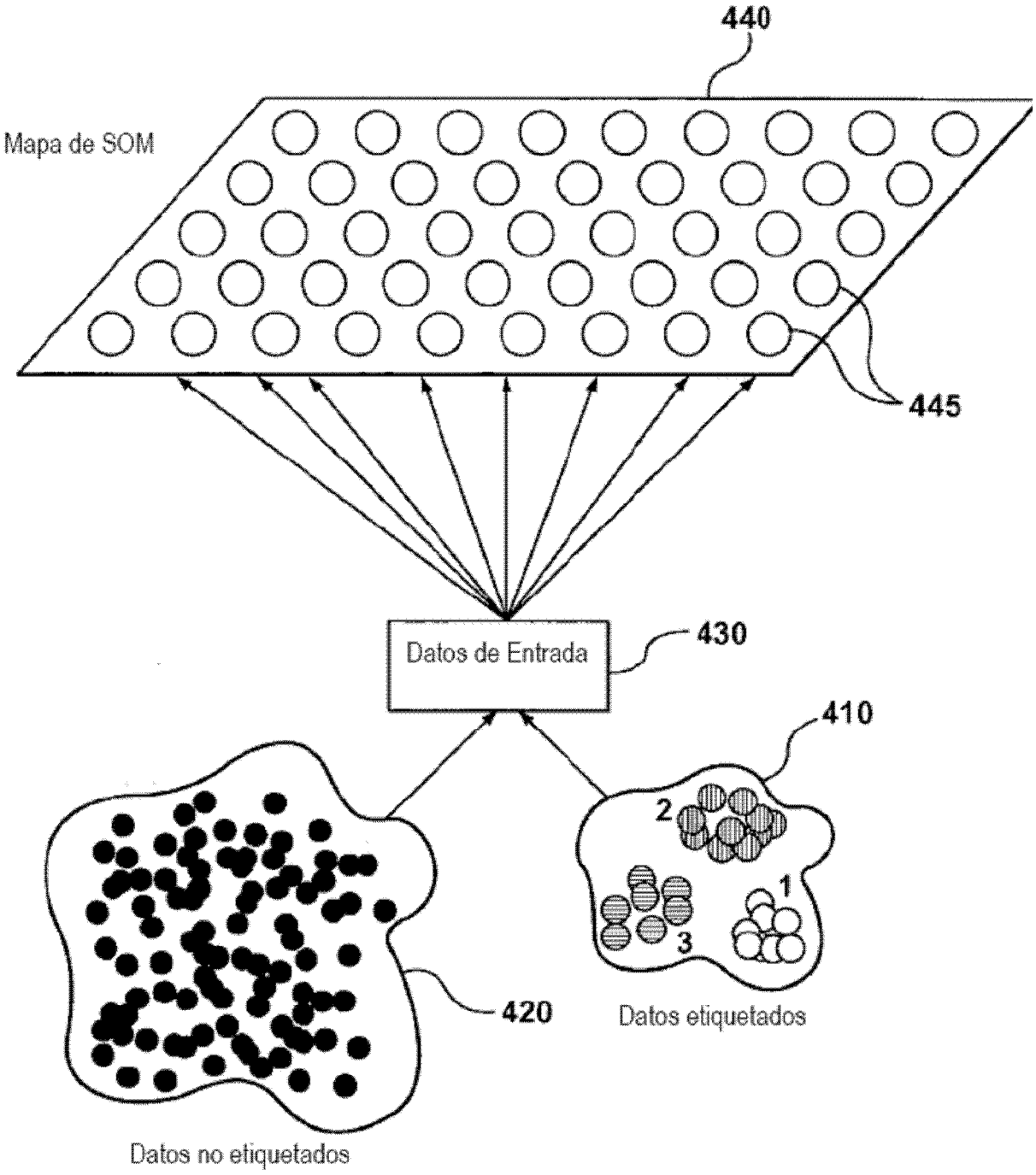
440
Mapa de SOM
445
Datos de Entrada
430
410
2
1
3
420
Datos etiquetados
Datos no etiquetados

FIG. 4

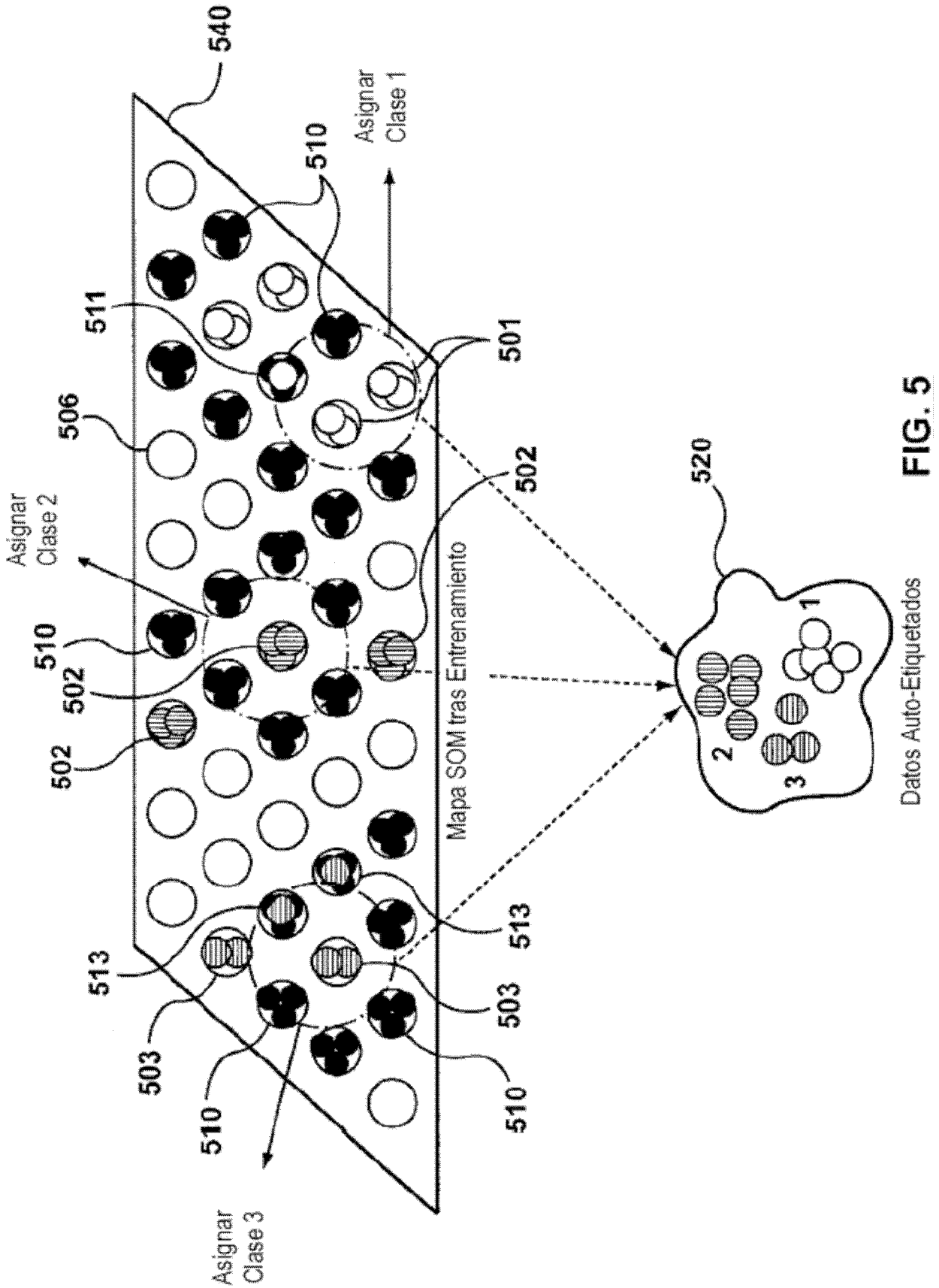
540
510
Asignar
Clase 1
511
501
506
Asignar
Clase 2
510
502
502
520
502
Mapa SOM tras Entrenamiento
513
513
503
503
510
510
Asignar
Clase 3
1
2
3
Datos Auto-Etiquetados

FIG. 5

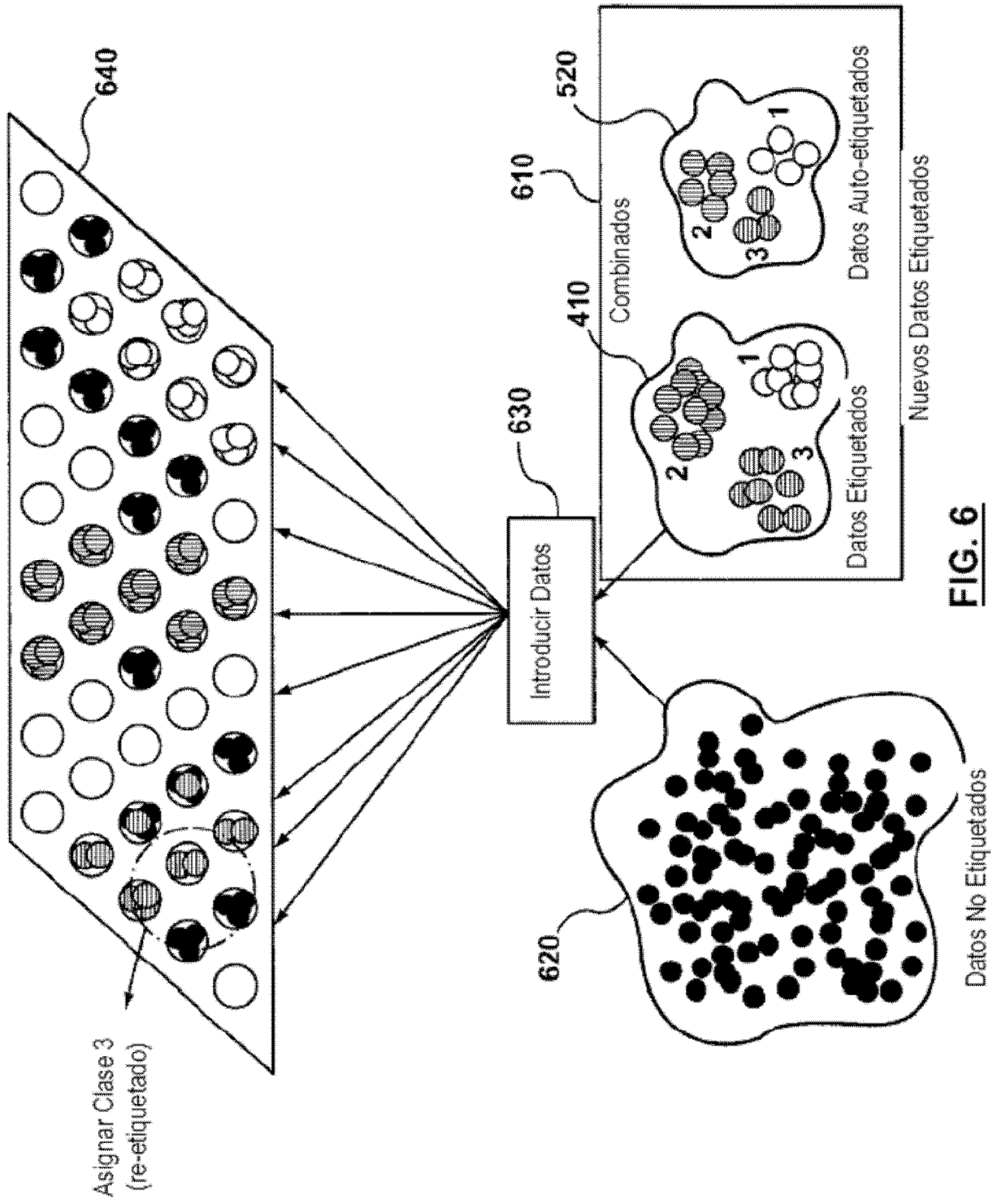
640
Asignar Clase 3
(re-etiquetado)
620
Datos No Etiquetados
630
Introducir Datos
610
520
Combinados
410
Datos Etiquetados
Datos Auto-etiquetados
Nuevos Datos Etiquetados
1
2
3

FIG. 6

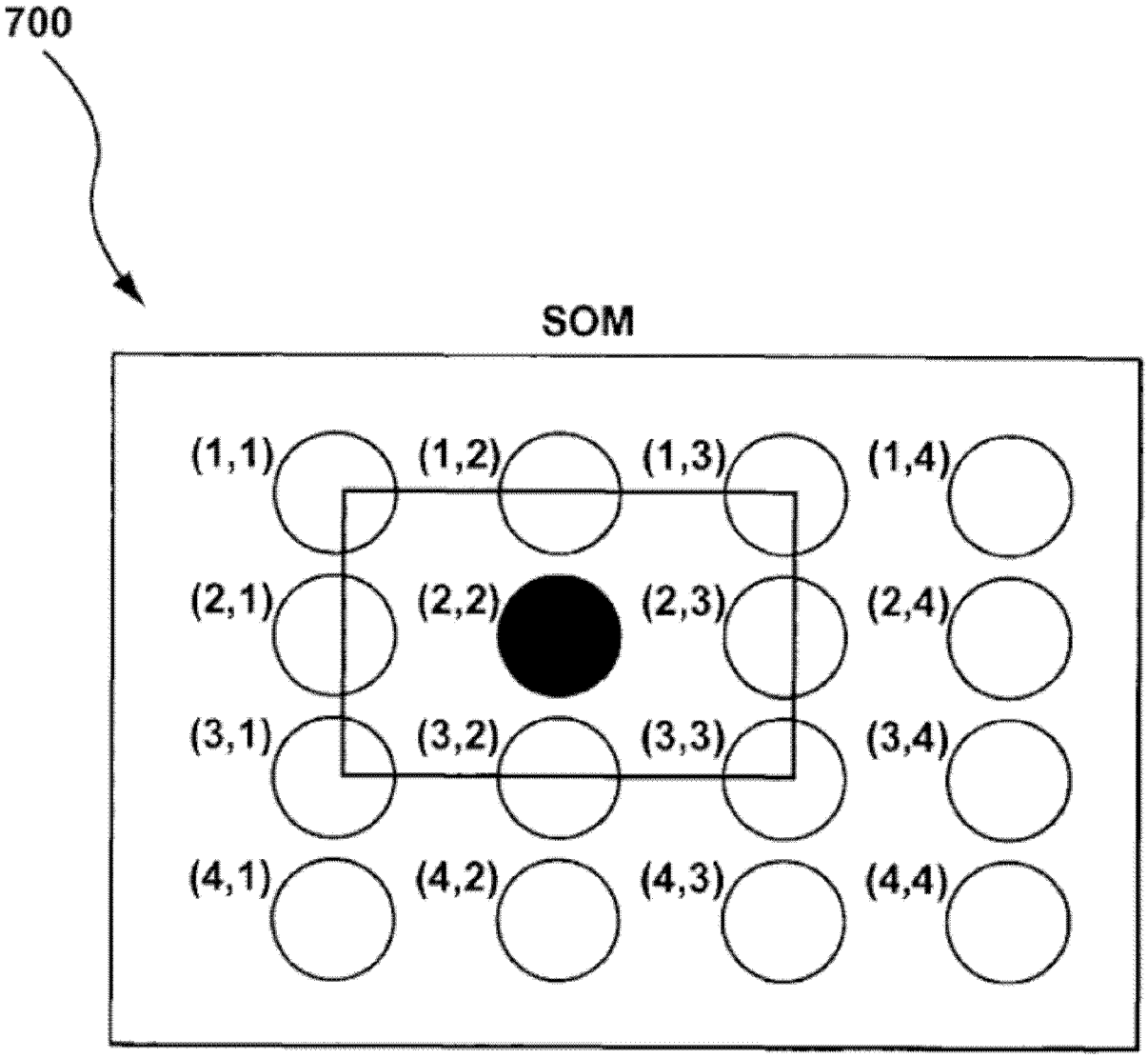

FIG. 7

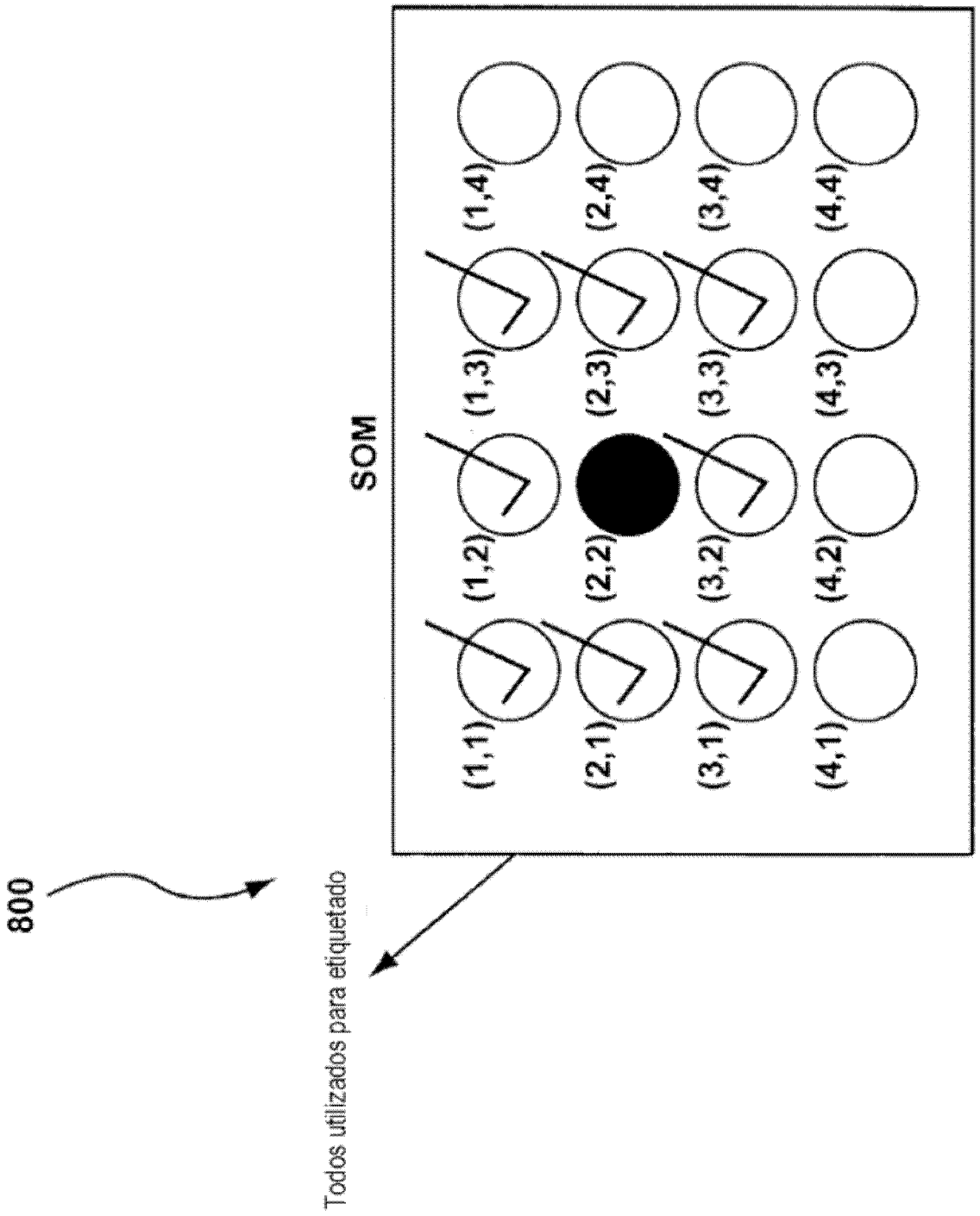
800
Todos utilizados para etiquetado
SOM
(1,1)
(1,2)
(1,3)
(1,4)
(2,1)
(2,2)
(2,3)
(2,4)
(3,1)
(3,2)
(3,3)
(3,4)
(4,1)
(4,2)
(4,3)
(4,4)

FIG. 8

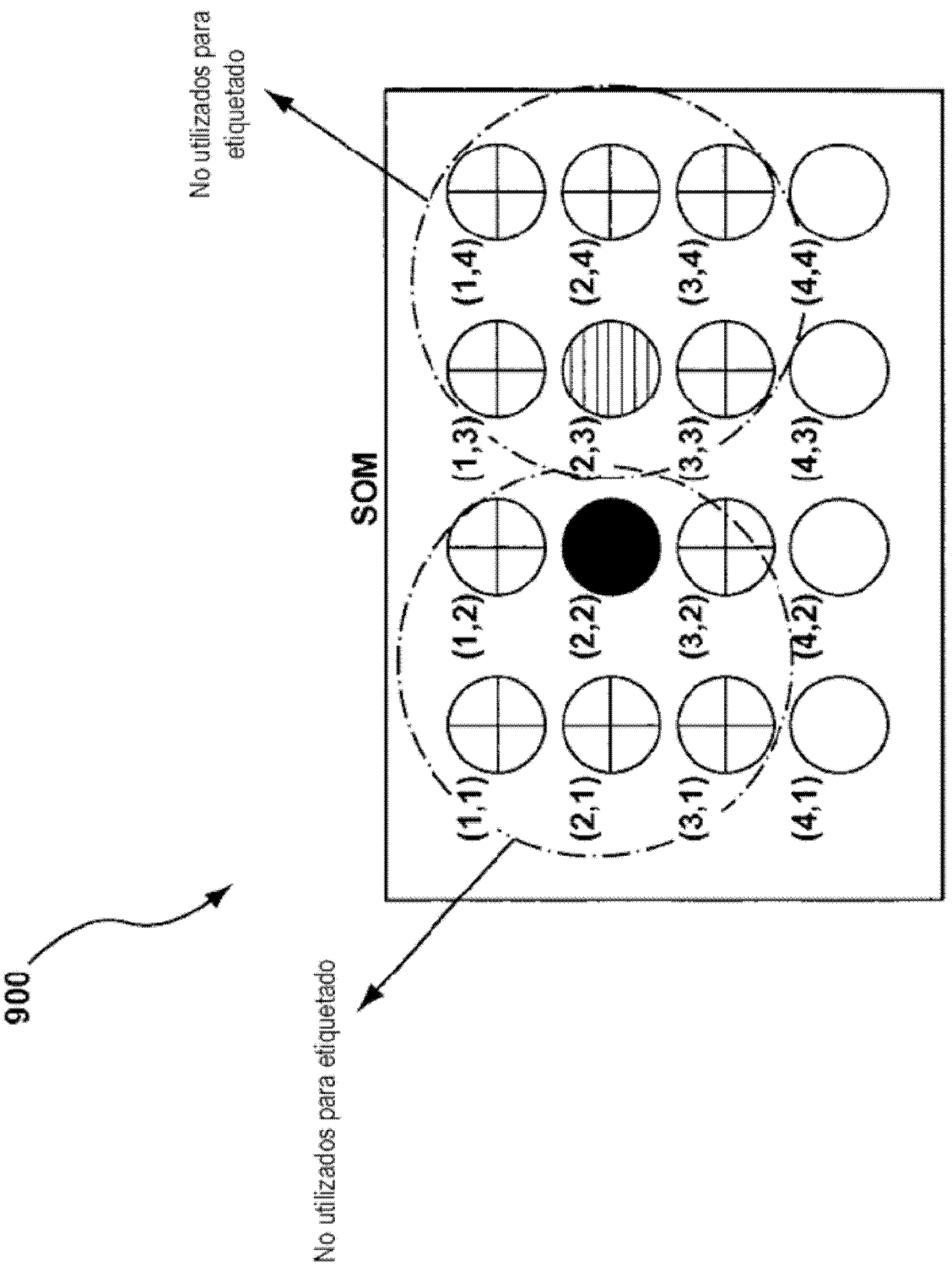
900
No utilizados para etiquetado
No utilizados para etiquetado
SOM
(1,1)
(2,1)
(3,1)
(4,1)
(1,2)
(2,2)
(3,2)
(4,2)
(1,3)
(2,3)
(3,3)
(4,3)
(1,4)
(2,4)
(3,4)
(4,4)

FIG. 9

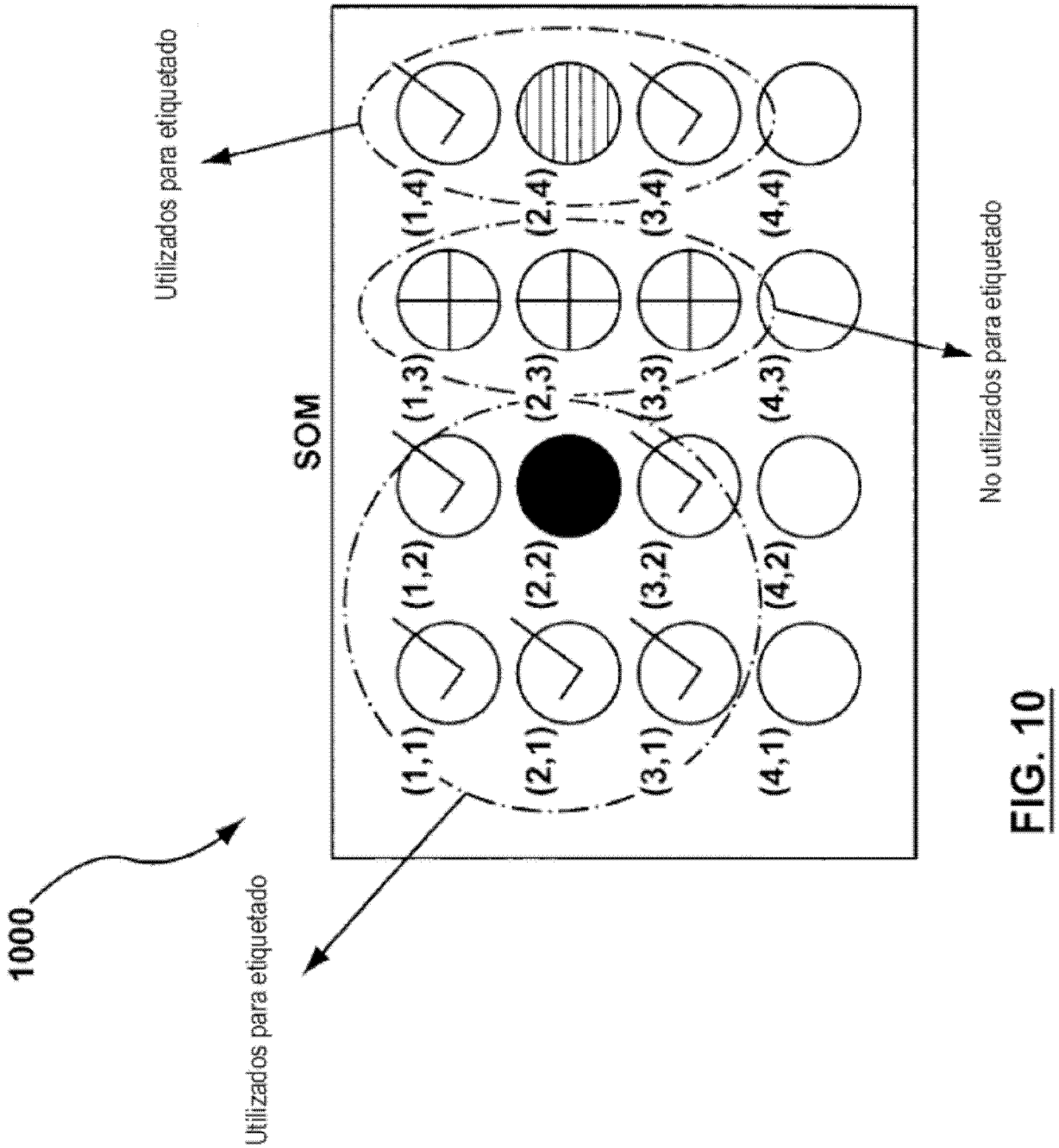
1000
SOM
Utilizados para etiquetado
Utilizados para etiquetado
No utilizados para etiquetado
(1,1)
(1,2)
(1,3)
(1,4)
(2,1)
(2,2)
(2,3)
(2,4)
(3,1)
(3,2)
(3,3)
(3,4)
(4,1)
(4,2)
(4,3)
(4,4)

FIG. 10

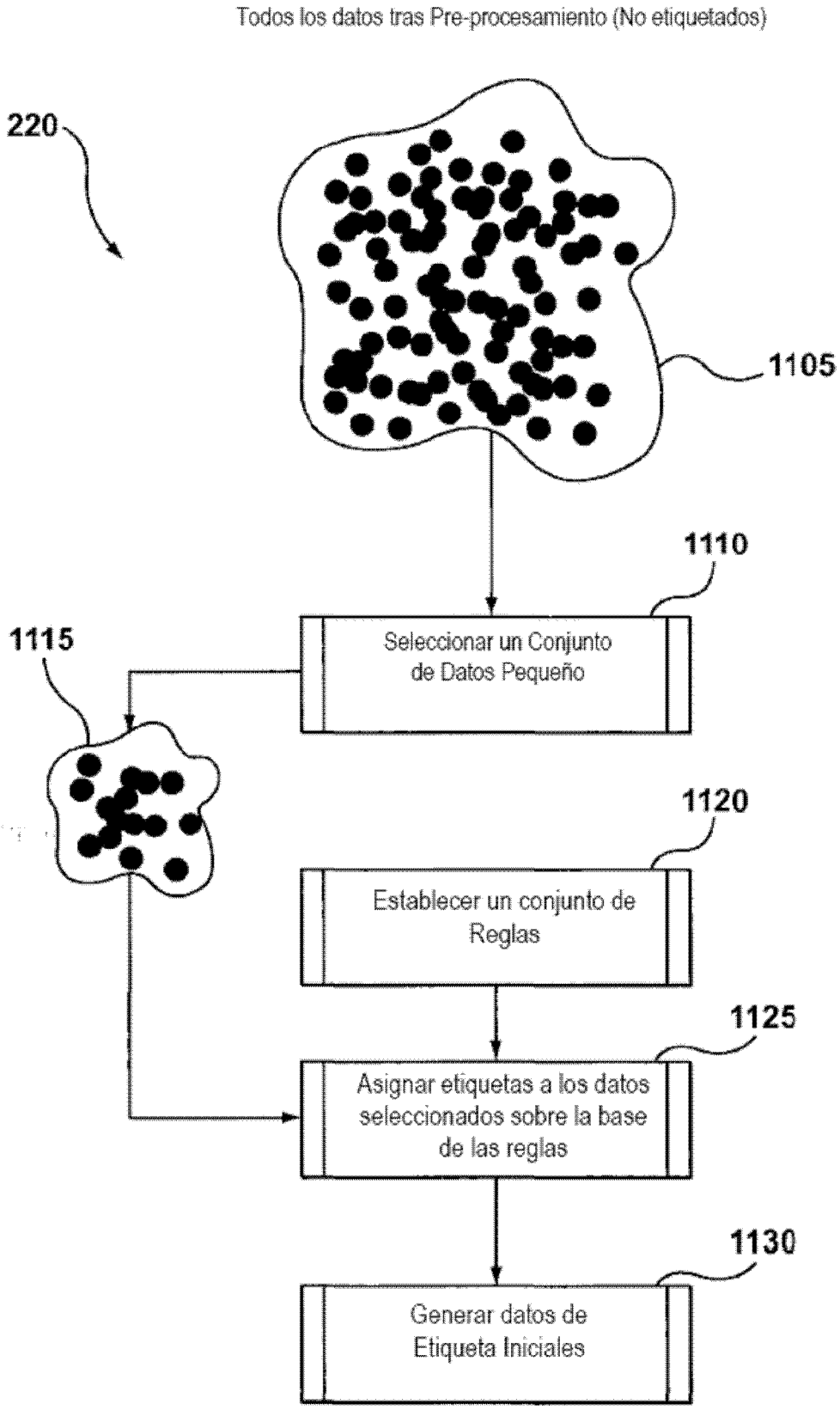
Todos los datos tras Pre-procesamiento (No etiquetados)
220
1105
1110
Seleccionar un Conjunto de Datos Pequeño
1115
1120
Establecer un conjunto de Reglas
1125
Asignar etiquetas a los datos seleccionados sobre la base de las reglas
1130
Generar datos de Etiqueta Iniciales

FIG. 11