(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 967 250**

(51) Int. Cl.:

G06N 20/00 (2009.01)
G06F 21/56 (2013.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **20.05.2019 PCT/CN2019/087564**

(87) Fecha y número de publicación internacional: **26.12.2019 WO19242444**

(96) Fecha de presentación y número de la solicitud europea: **20.05.2019 E 19821642 (6)**

(97) Fecha y número de publicación de la concesión europea: **20.09.2023 EP 3812975**

(54) Título: **Método y sistema para entrenar un motor de aprendizaje automático y dispositivo relacionado**

(30) Prioridad:

20.06.2018 CN 201810638258

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
29.04.2024

(73) Titular/es:

SANGFOR TECHNOLOGIES INC. (100.0%)
Block A1, Nanshan Ipark, No.1001 Xueyuan Road, Nanshan District
Shenzhen, Guangdong 518055, CN

(72) Inventor/es:

WEI, KAIZHI y
ZHANG, MINGXING

(74) Agente/Representante:

ARIAS SANZ, Juan

DESCRIPCIÓN

Método y sistema para entrenar un motor de aprendizaje automático y dispositivo relacionado

La presente solicitud reivindica prioridad con respecto a la solicitud de patente china n.º 201810638258.2, titulada "METHOD AND SYSTEM FOR TRAINING MACHINE LEARNING ENGINE AND RELATED DEVICE", presentada el 20 de junio de 2018 con la Administración Nacional de Propiedad Intelectual de China.

Campo

La presente divulgación se refiere al campo del aprendizaje automático, y en particular a un método y un sistema para entrenar un motor de aprendizaje automático, un medio de almacenamiento legible por ordenador y un dispositivo para entrenar un motor de aprendizaje automático.

Antecedentes

Actualmente, la tecnología de las grandes cantidades de datos (del inglés, "big data") y de aprendizaje automático (del inglés, "machine learning") es un área de investigación destacada en la industria y el mundo académico. La tecnología de aprendizaje automático y grandes cantidades de datos se aplica cada vez más a la seguridad de información, logrando buenos resultados. La tecnología de detección de *software* malicioso (del inglés, "malware") basada en el aprendizaje automático ha avanzado mucho, y un gran número de compañías, tal como Cylance, que proporcionan productos de detección de *software* malicioso basados en el aprendizaje automático ha surgido en la industria. Las empresas emergentes se ven respaldadas por VC y han logrado muchos buenos resultados.

En los motores de aprendizaje automático convencionales, la entropía de información de un *software* se determina como características estáticas para el aprendizaje automático. Generalmente, la entropía de información se extrae de las características principales de un archivo ejecutable, y la entropía de información extraída se usa como características estáticas para ser introducidas en el motor de aprendizaje automático. Sin embargo, la operación de extraer la entropía de información de las características principales del archivo ejecutable solo se realiza en una tabla de importación, una tabla de exportación e información de sección del archivo ejecutable, la tabla de importación, la tabla de exportación y la información de sección son características generales, por lo tanto, las características de muestra extraídas tienen granularidad gruesa. El aprendizaje automático basado en las características de muestra según la tecnología convencional dará como resultado un efecto deficiente en el entrenamiento del motor de aprendizaje automático y una baja tasa de reconocimiento.

Por lo tanto, cómo extraer de manera estable las características estáticas del archivo ejecutable para hacer que el modelo de detección generado por el motor de aprendizaje automático tenga una tasa de reconocimiento con fuerte robustez es un problema que actualmente se requiere resolver por los expertos en la técnica. El documento de patente US2017004306 representa el estado de la técnica anterior más cercano.

Sumario

Un método y un sistema para entrenar un motor de aprendizaje automático, se proporcionan un medio de almacenamiento legible por ordenador y un dispositivo para entrenar un motor de aprendizaje automático según la presente divulgación, extraer de manera estable características estáticas de un archivo ejecutable para hacer que un modelo de detección generado por un motor de aprendizaje automático tenga una tasa de reconocimiento con fuerte robustez.

Para resolver los problemas técnicos anteriores, se proporciona un método para entrenar un motor de aprendizaje automático según la presente divulgación. El método incluye:

obtener un archivo ejecutable, y dividir el archivo ejecutable en múltiples partes a extraer basándose en un directorio de datos del archivo ejecutable;

extraer características de entropía de información de todas las partes a extraer, y realizar la transformación de datos sobre las características de entropía de información para obtener características estáticas; y

entrenar un motor de aprendizaje automático usando las características estáticas.

La división del archivo ejecutable en múltiples partes a extraer basándose en un directorio de datos del archivo ejecutable incluye:

dividir el archivo ejecutable en una parte de encabezado de archivo ejecutable, una parte de sección de archivo ejecutable, una parte de datos de superposición (del inglés, "overlay") de archivo ejecutable, y una parte integral de archivo ejecutable basándose en el directorio de datos del archivo ejecutable.

En una realización, la extracción de características de entropía de información de todas las partes a extraer y la realización de transformación de datos sobre las características de entropía de información para obtener características estáticas incluyen:

extraer las características de entropía de información de todas las partes a extraer, donde las características de entropía de información son datos binarios;

calcular una proporción de cada uno de los caracteres en los datos binarios correspondientes a las características de entropía de información con respecto a todos los caracteres en los datos binarios, y obtener un flujo de entropía de las características de entropía de información basándose en la proporción; y

realizar una transformada DCT o una transformada DWT sobre el flujo de entropía para obtener las características estáticas.

En una realización, el cálculo de una proporción de cada uno de los caracteres en los datos binarios correspondientes a las características de entropía de información con respecto a todos los caracteres en los datos binarios y obtener un flujo de entropía de las características de entropía de información basándose en la proporción incluye:

dividir los datos binarios correspondientes a las características de entropía de información en una pluralidad de bloques de bytes, donde cada uno de los bloques de bytes incluye 256 bytes;

para cada uno de los bloques de bytes, calcular una proporción de bloques de bytes p de cada uno de los caracteres en el bloque de bytes con respecto a todos los caracteres en el bloque de bytes, y calcular una componente de flujo de entropía S para cada uno de los bytes en el bloque de bytes usando la siguiente ecuación:

$$S = \left| p \cdot \log(p) \right|; \text{ y}$$

acumular todas las componentes de flujo de entropía para obtener el flujo de entropía.

En una realización, después de obtener el archivo ejecutable, el método incluye además:

determinar un formato de archivo del archivo ejecutable basándose en información de formato general del archivo ejecutable; y seleccionar una estrategia de extracción de características basándose en el formato de archivo.

La extracción de características de entropía de información de todas las partes a extraer incluye: extraer las características de entropía de información de todas las partes a extraer basándose en la estrategia de extracción de características.

En una realización, la determinación de un formato de archivo del archivo ejecutable basándose en información de formato general del archivo ejecutable incluye:

determinar valores de RVA de directorio de metadatos .Net (del inglés, ".Net MetaData Directory RVA") y tamaño de directorio de metadatos .Net (del inglés, ".Net MetaData Directory Size") en el directorio de datos del archivo ejecutable; y

determinar, si los valores de la RVA de directorio de metadatos .Net y el tamaño de directorio de metadatos .Net no están vacíos, que el formato de archivo del archivo ejecutable es formato .Net.

En una realización, la división del archivo ejecutable en múltiples partes a extraer basándose en un directorio de datos del archivo ejecutable incluye:

dividir el archivo ejecutable en múltiples partes a extraer basándose en una matriz de secciones del archivo ejecutable.

En una realización, después de entrenar un motor de aprendizaje automático usando las características estáticas, el método incluye además:

extraer, en el caso de que se detecte un *software* sospechoso, un archivo ejecutable objetivo del *software* sospechoso; e

introducir el archivo ejecutable objetivo en el motor de aprendizaje automático entrenado para determinar, mediante el motor de aprendizaje automático entrenado, si el archivo ejecutable objetivo es un archivo ejecutable de un *software* malicioso.

En una realización, el método incluye además:

informar, si se determina mediante el motor de aprendizaje automático entrenado que el archivo ejecutable objetivo es el archivo ejecutable del *software* malicioso, de información de *software* del *software* sospechoso para realizar la detección manual sobre el *software* sospechoso;

recibir un resultado de detección manual, y determinar si la determinación actual mediante el motor de aprendizaje automático entrenado es errónea basándose en el resultado de detección manual; y

entrenar, si la determinación actual mediante el motor de aprendizaje automático entrenado es errónea, el motor de aprendizaje automático usando el archivo ejecutable objetivo como una nueva muestra de entrenamiento negativo.

Se proporciona además un sistema para entrenar un motor de aprendizaje automático según la presente divulgación. El sistema incluye un módulo de división de archivo, un módulo de extracción de características estáticas y un módulo de entrenamiento.

El módulo de división de archivo está configurado para obtener un archivo ejecutable y dividir el archivo ejecutable en múltiples partes a extraer basándose en un directorio de datos del archivo ejecutable.

El módulo de extracción de características estáticas está configurado para extraer características de entropía de información de todas las partes a extraer y realizar la transformación de datos sobre las características de entropía de información para obtener características estáticas.

El módulo de entrenamiento está configurado para entrenar un motor de aprendizaje automático usando las características estáticas.

El módulo de división de archivo incluye una unidad de obtención de archivos y una unidad de división.

La unidad de obtención de archivos está configurada para obtener el archivo ejecutable.

La unidad de división está configurada para dividir el archivo ejecutable en una parte de encabezado de archivo ejecutable, una parte de sección de archivo ejecutable, una parte de datos de superposición de archivo ejecutable, y una parte integral de archivo ejecutable basándose en el directorio de datos del archivo ejecutable.

En una realización, el módulo de extracción de características estáticas incluye una unidad de extracción de características de entropía de información, una unidad de cálculo de flujo de entropía y una unidad de transformación de datos.

La unidad de extracción de características de entropía de información está configurada para extraer las características de entropía de información de todas los partes a extraer. La característica de entropía de información son datos binarios.

La unidad de cálculo de flujo de entropía está configurada para calcular una proporción de cada uno de los caracteres en los datos binarios correspondientes a las características de entropía de información con respecto a todos los caracteres en los datos binarios y obtener un flujo de entropía de las características de entropía de información basándose en la proporción.

La unidad de transformación de datos está configurada para realizar una transformada DCT o una transformada DWT sobre el flujo de entropía para obtener las características estáticas.

En una realización, la unidad de cálculo de flujo de entropía incluye una unidad secundaria de división, una unidad secundaria de cálculo de componente de flujo de entropía y una unidad de acumulación de componentes de flujo de entropía.

La unidad secundaria de división está configurada para dividir los datos binarios correspondientes a las características de entropía de información en múltiples bloques de bytes, donde cada uno de los bloques de bytes incluye 256 bytes.

La unidad secundaria de cálculo de componente de flujo de entropía está configurada para, para cada uno de los bloques de bytes, calcular una proporción de bloques de bytes p de cada uno de los caracteres en el bloque de bytes con respecto a todos los caracteres en el bloque de bytes, y calcular una componente de flujo de entropía S para cada uno de los bytes en el bloque de bytes usando la siguiente ecuación:

$$S = \left| p \cdot \log (p) \right|.$$

La unidad de acumulación de componentes de flujo de entropía está configurada para acumular todas las componentes de flujo de entropía para obtener el flujo de entropía.

En una realización, el sistema incluye además un módulo de determinación de estrategia de extracción.

El módulo de determinación de estrategia de extracción está configurado para determinar un formato de archivo del archivo ejecutable basándose en la información de formato general del archivo ejecutable, y seleccionar una estrategia de extracción de características basándose en el formato de archivo.

El módulo de extracción de características estáticas está configurado para extraer las características de entropía de información de todas las partes a extraer basándose en la estrategia de extracción de características, y realizar la transformación de datos sobre las características de entropía de información para obtener las características estáticas.

En una realización, el módulo de determinación de estrategia de extracción incluye una unidad de determinación de formato de archivo y una unidad de selección de estrategia.

La unidad de determinación de formato de archivo está configurada para determinar valores de RVA de directorio de metadatos .Net y tamaño de directorio de metadatos .Net en el directorio de datos del archivo ejecutable, y determinar, si los valores de la RVA de directorio de metadatos .Net y el tamaño de directorio de metadatos .Net no están vacíos, que el formato de archivo del archivo ejecutable es formato .Net.

La unidad de selección de estrategia está configurada para seleccionar la estrategia de extracción de características basándose en el formato de archivo.

En una realización, el módulo de división de archivo está configurado para obtener el archivo ejecutable y dividir el archivo ejecutable en las múltiples partes a extraer basándose en una matriz de secciones del archivo ejecutable.

En una realización, el sistema incluye además un módulo de extracción y un módulo de determinación.

El módulo de extracción está configurado para extraer, en el caso de que se detecte un *software* sospechoso, un archivo ejecutable objetivo del *software* sospechoso.

El módulo de determinación está configurado para introducir el archivo ejecutable objetivo en el motor de aprendizaje automático entrenado para determinar, mediante el motor de aprendizaje automático entrenado, si el archivo ejecutable objetivo es un archivo ejecutable de un *software* malicioso.

En una realización, el sistema incluye además un módulo de informe y un módulo de entrenamiento secundario.

El módulo de informe está configurado para informar, si se determina mediante el motor de aprendizaje automático entrenado que el archivo ejecutable objetivo es el archivo ejecutable del *software* malicioso, de información de *software* del *software* sospechoso para realizar la detección manual sobre el *software* sospechoso.

El módulo de entrenamiento secundario está configurado para recibir un resultado de detección manual, determinar si la determinación actual mediante el motor de aprendizaje automático entrenado es errónea basándose en el resultado de detección manual, y entrenar, si la determinación actual mediante el motor de aprendizaje automático entrenado es errónea, el motor de aprendizaje automático usando el archivo ejecutable objetivo como una nueva muestra de entrenamiento negativo.

Se proporciona además un medio de almacenamiento legible por ordenador que almacena un programa informático según la presente divulgación. El programa informático, cuando se ejecuta, realiza el método para entrenar un motor de aprendizaje automático.

Se proporciona además un dispositivo para entrenar un motor de aprendizaje automático según la presente divulgación. El dispositivo incluye una memoria y un procesador. La memoria almacena un programa informático. El procesador está configurado para ejecutar el programa informático para realizar el método para entrenar un motor de aprendizaje automático.

Se proporciona un método para entrenar un motor de aprendizaje automático según la presente divulgación. El método incluye: obtener un archivo ejecutable y dividir el archivo ejecutable en múltiples partes a extraer basándose en un directorio de datos del archivo ejecutable; extraer características de entropía de información de todas las partes a extraer y realizar transformación de datos sobre las características de entropía de información para obtener características estáticas; y entrenar un motor de aprendizaje automático usando las características estáticas.

En la presente divulgación, el archivo ejecutable se divide en múltiples partes a extraer, se extraen características de entropía de información de todas las partes a extraer, se realiza una transformación sobre las características de entropía de información extraídas para obtener características estáticas que se requieren al entrenar el motor de

aprendizaje automático para entrenar el motor de aprendizaje automático. En comparación con el método según la tecnología convencional en la que solo se extrae la entropía de información de las características principales del archivo ejecutable, con el método en la presente divulgación, el archivo ejecutable se divide en múltiples partes a extraer basándose en el directorio de datos y se extraen las características de entropía de información de todas las partes a extraer. Según la tecnología convencional, dado que las características de entropía de información se extraen de las características principales que son características comunes, las características estáticas obtenidas tienen una granularidad gruesa. Según la presente divulgación, las características de entropía de información se extraen del archivo ejecutable completo dividiendo el archivo ejecutable en múltiples partes a extraer y extrayendo las características de entropía de información de todas las partes a extraer, por lo tanto, se puede garantizar que las características estáticas obtenidas tienen una granularidad fina. Con el método según la presente divulgación, las características estáticas del archivo ejecutable se pueden extraer de manera estable, de modo que el modelo de detección generado por el motor de aprendizaje automático tiene una tasa de reconocimiento con fuerte robustez. Según la presente divulgación, un sistema para entrenar un motor de aprendizaje automático, se proporcionan además un medio de almacenamiento legible por ordenador y un dispositivo para entrenar un motor de aprendizaje automático, que tienen los efectos beneficiosos anteriores y no se repiten en el presente documento.

Breve descripción de los dibujos

Con el fin de ilustrar las realizaciones de la presente divulgación o soluciones técnicas en la tecnología convencional más claramente, los dibujos que se utilizarán en la descripción de las realizaciones o la tecnología convencional se introducen simplemente a continuación. Es evidente que los dibujos descritos a continuación solo ilustran algunas realizaciones de la presente divulgación. Para los expertos en la técnica, se pueden obtener otros dibujos según estos dibujos sin ningún trabajo creativo.

La figura 1 es un diagrama de flujo de un método para entrenar un motor de aprendizaje automático según una realización de la presente divulgación;

la figura 2 es un diagrama de flujo de extracción de características de entropía de información y realización de transformación sobre las características de entropía de información extraídas para obtener características estáticas en un método para entrenar un motor de aprendizaje automático según una realización de la presente divulgación;

la figura 3 es un diagrama de flujo de un método para entrenar un motor de aprendizaje automático según otra realización de la presente divulgación;

la figura 4 es un diagrama de flujo de un método para entrenar un motor de aprendizaje automático según otra realización de la presente divulgación; y

la figura 5 es un diagrama estructural esquemático de un sistema para entrenar un motor de aprendizaje automático según una realización de la presente divulgación.

Descripción detallada

Con el fin de hacer que los objetivos, soluciones técnicas y ventajas de las realizaciones de la presente divulgación estén más claras, soluciones técnicas según las realizaciones de la presente divulgación se describen de manera clara y completa a continuación en el presente documento junto con los dibujos usados en las realizaciones de la presente divulgación. Es evidente que las realizaciones descritas son solo algunas realizaciones de la presente divulgación en lugar de todas las realizaciones. Todas las demás realizaciones obtenidas por los expertos en la técnica basadas en las realizaciones de la presente divulgación sin ningún trabajo creativo caen dentro del alcance de protección de la presente divulgación.

Se hace referencia a la figura 1, que es un diagrama de flujo de un método para entrenar un motor de aprendizaje automático según una realización de la presente divulgación.

El método incluye las siguientes etapas S101 a S103.

En la etapa S101, se obtiene un archivo ejecutable, y el archivo ejecutable se divide en múltiples partes a extraer basándose en un directorio de datos del archivo ejecutable.

En la realización, se proporciona un método para entrenar un motor de aprendizaje automático. Las muestras requeridas para entrenar el motor de aprendizaje automático son características estáticas del archivo ejecutable. El archivo ejecutable es un archivo que puede cargarse y ejecutarse por un sistema operativo. En diferentes entornos de sistemas operativos, el archivo ejecutable está en diferentes formatos. Por ejemplo, el archivo ejecutable en un sistema Windows es un archivo PE (ejecutable portable, del inglés "Portable Executable"), y el archivo ejecutable en un sistema, tales como Linux, Mac y Android, tiene un formato de archivo correspondiente al sistema. El sistema donde se ejecuta el archivo ejecutable no está limitado en el presente documento. Según la presente divulgación, las operaciones se pueden realizar sobre los archivos ejecutables en múltiples sistemas.

En esta etapa, el archivo ejecutable se divide en múltiples partes a extraer basándose en el directorio de datos del archivo ejecutable. Específicamente, el archivo ejecutable puede dividirse basándose en una matriz de secciones en la estructura del archivo ejecutable, y puede dividirse basándose en otros directorios de datos en la estructura del archivo ejecutable, que no se limita en el presente documento. El archivo ejecutable se divide en una parte de encabezado de archivo ejecutable, una parte de sección de archivo ejecutable, una parte de datos de superposición (del inglés "Overlay data part") de archivo ejecutable, y una parte integral de archivo ejecutable basándose en el directorio de datos del archivo ejecutable. En una realización, después de determinar la información de formato general del archivo ejecutable, el archivo ejecutable puede dividirse en las partes a extraer basándose en algunas partes o algunas estructuras de la información de formato general del archivo ejecutable.

En esta etapa, el archivo ejecutable se divide en múltiples partes a extraer basándose en el directorio de datos. Para cada una de las partes a extraer, las características de entropía de información pueden extraerse con un método de extracción de características específicas, logrando características de entropía de información con una granularidad fina, aumentando de ese modo la estabilidad de las características estáticas.

En la etapa S102, se extraen características de entropía de información de todas las partes a extraer, y la transformación de datos se realiza sobre las características de entropía de información para obtener características estáticas.

En esta etapa, basándose en que el archivo ejecutable se ha dividido en múltiples partes a extraer en la etapa S101, se extraen las características de entropía de información de todas las partes a extraer. La entropía de información es una probabilidad de presencia de una clase de información, y puede compararse con la entropía termodinámica. Las características de entropía de información en la realización son probabilidades de presencia de todas las clases de información en las partes a extraer. Generalmente, la información en un ordenador es en forma de caracteres. Las características de entropía de información en esta etapa pueden considerarse como valores de características de las probabilidades de presencia de los caracteres en el archivo ejecutable.

Debe entenderse que, las muestras usadas para entrenar el motor de aprendizaje automático son características estáticas de los archivos ejecutables. En esta etapa, la transformación de datos se realiza sobre las características de entropía de información para obtener las características estáticas, de modo que el motor de aprendizaje automático se entrena en una siguiente etapa.

En la etapa S103, el motor de aprendizaje automático se entrena usando las características estáticas.

En la realización, el archivo ejecutable se divide en múltiples partes a extraer, se extraen características de entropía de información de todas las partes a extraer, y la transformación se realiza sobre las características de entropía de información extraídas para obtener características estáticas que se requieren al entrenar el motor de aprendizaje automático para entrenar el motor de aprendizaje automático. En comparación con el método según la tecnología convencional en el que solo se extrae la entropía de información de las características principales del archivo ejecutable, con el método en la realización, el archivo ejecutable se divide en múltiples partes a extraer basándose en el directorio de datos y se extraen características de entropía de información de todas las partes a extraer. Según la tecnología convencional, dado que las características de entropía de información se extraen de las características principales que son características comunes, las características estáticas obtenidas tienen una granularidad gruesa. Según la realización, las características de entropía de información se extraen del archivo ejecutable completo dividiendo el archivo ejecutable en múltiples partes a extraer y extrayendo las características de entropía de información de todas las partes a extraer, por lo tanto, se puede garantizar que las características estáticas obtenidas tienen una granularidad fina. Con el método según la realización, las características estáticas del archivo ejecutable se pueden extraer de manera estable, de modo que el modelo de detección generado por el motor de aprendizaje automático tiene una tasa de reconocimiento con fuerte robustez.

Se hace referencia a la figura 2, que es un diagrama de flujo de extracción de características de entropía de información y realización de transformación sobre las características de entropía de información extraídas para obtener características estáticas en un método para entrenar un motor de aprendizaje automático según una realización de la presente divulgación.

La extracción de características de entropía de información y la realización de transformación sobre las características de entropía de información extraídas para obtener características estáticas incluye, pero no se limita a, las siguientes etapas.

En la etapa S201, se extraen las características de entropía de información de todas las partes a extraer. Las características de entropía de información son datos binarios.

En la etapa S202, se calcula una proporción de cada uno de los caracteres en los datos binarios correspondientes a las características de entropía de información con respecto a todos los caracteres en los datos binarios, y se obtiene un flujo de entropía de las características de entropía de información basándose en la proporción.

En el método según la presente divulgación, la entropía de información puede compararse con la entropía termodinámica, y la transmisión de entropía neta a través del flujo de información se denomina flujo de entropía.

En esta etapa, el flujo de entropía se calcula basándose en las características de entropía de información global que son datos binarios. En una realización, los datos binarios pueden dividirse en múltiples bloques para calcular el flujo de entropía. Por ejemplo, los datos binarios correspondientes a las características de entropía de información se dividen en múltiples bloques de bytes, donde cada uno de los bloques de bytes incluye 256 bytes. Para cada uno de los bloques de bytes, se calcula una proporción de bloques de bytes p de cada uno de los caracteres en el bloque de bytes con respecto a todos los caracteres en el bloque de bytes, y se calcula una componente de flujo de entropía S para cada uno de los bytes en el bloque de bytes usando la ecuación:

$$S = \left| p \cdot \log(p) \right|$$

Todas las componentes de flujo de entropía se acumulan para obtener el flujo de entropía.

En la etapa S203, se realiza una transformada DCT o una transformada DWT sobre el flujo de entropía para obtener las características estáticas.

La DCT (transformada discreta de coseno) es una transformada relacionada con la transformada de Fourier. La DWT (transformada discreta de Walsh) es una transformada ortogonal no sinusoidal con una función de Walsh como función básica. Las características estáticas pueden obtenerse realizando la transformada DCT o la transformada DWT sobre el flujo de entropía.

Se hace referencia a la figura 3, que es un diagrama de flujo de un método para entrenar un motor de aprendizaje automático según otra realización de la presente divulgación.

El método incluye las siguientes etapas S301 a S303.

En la etapa S301, un formato de archivo del archivo ejecutable se determina basándose en la información de formato general del archivo ejecutable, y se selecciona una estrategia de extracción de características basándose en el formato de archivo.

En esta etapa, una subclase (tal como, formato .Net, un formato Py2exe, un formato auotit y un formato Nsis) del archivo ejecutable se determina basándose en la información de formato general del archivo ejecutable, y se determina una estrategia de extracción de características (tal como una estrategia para determinar qué características se extraerán) basándose en el formato de archivo del archivo ejecutable.

Los siguientes son ejemplos para determinar el formato de archivo. Al determinar el formato de archivo del archivo ejecutable, se puede determinar si los valores de "RVA de directorio de metadatos .Net" y "tamaño de directorio de metadatos .Net" en el directorio de datos del archivo ejecutable están vacíos. Si los valores del "RVA de directorio de metadatos .Net" y el "tamaño de directorio de metadatos .Net" en el directorio de datos del archivo ejecutable no están vacíos, el archivo ejecutable está en formato .Net. Alternativamente, se puede detectar si el archivo ejecutable incluye una referencia de una biblioteca mientras se ejecuta *Python*. Por ejemplo, se detecta si una tabla de importación del archivo ejecutable incluye formato *Python**.dll u otros formatos (tanto si se incluyen códigos compilados con códigos *Python* o *Python*) para determinar si el archivo ejecutable está en formato Py2exe. Los archivos ejecutables en diferentes formatos de archivo pueden determinarse de diferentes maneras, que no están limitadas aquí y pueden configurarse por los expertos en la técnica según las condiciones de aplicación reales.

En la etapa S302, las características estáticas del archivo ejecutable se extraen basándose en la estrategia de extracción de características.

Aunque las estrategias de extracción de características son diferentes, el propósito de extraer las características estáticas puede ser el siguiente: extraer determinadas estructuras y metadatos del archivo ejecutable, leer y analizar los atributos de cada una de las estructuras y metadatos según una estructura predefinida, convertir datos numéricos en un determinado intervalo, y realizar la operación, tales como interceptar caracteres de una determinada longitud o reemplazar caracteres especiales, en datos de cadena.

En la etapa S303, el motor de aprendizaje automático se entrena usando las características estáticas.

Con la solución técnica según la realización, se pueden obtener características estáticas que tienen una granularidad fina para entrenar al motor de aprendizaje automático, mejorar la tasa de reconocimiento del motor de aprendizaje automático. Las operaciones realizadas en la realización pueden combinarse con las operaciones realizadas en la primera realización o la segunda realización, de modo que las características de entropía de información en las dos

realizaciones anteriores se extraen basándose en la estrategia de extracción de características seleccionada basándose en el formato de archivo según la realización. Es decir, las características de entropía de información de las partes a extraer se extraen basándose en la estrategia de extracción de características.

Se hace referencia a la figura 4, que es un diagrama de flujo de un método para entrenar un motor de aprendizaje automático según otra realización de la presente divulgación.

En la etapa S401, se obtiene un archivo ejecutable, y el archivo ejecutable se divide en múltiples partes a extraer basándose en un directorio de datos del archivo ejecutable.

En la etapa S402, un formato de archivo del archivo ejecutable se determina basándose en la información de formato general del archivo ejecutable, y se selecciona una estrategia de extracción de características basándose en el formato de archivo.

En la etapa S403, las características de entropía de información de todas las partes a extraer se extraen basándose en la estrategia de extracción de características. Las características de entropía de información son datos binarios.

En la etapa S404, los datos binarios correspondientes a las características de entropía de información se dividen en múltiples bloques de bytes, donde cada uno de los bloques de bytes incluye 256 bytes.

En la etapa S405, para cada uno de los bloques de bytes, se calcula una proporción de bloques de bytes p de cada uno de los caracteres en el bloque de bytes con respecto a todos los caracteres en el bloque de bytes, y se calcula una componente de flujo de entropía S para cada uno de los bytes en el bloque de bytes usando la siguiente ecuación:

$$S = \left| p \cdot \log(p) \right|$$

En la etapa S406, todas las componentes de flujo de entropía se acumulan para obtener el flujo de entropía.

En la etapa S407, se realiza una transformada DCT o una transformada DWT sobre el flujo de entropía para obtener las características estáticas.

En la etapa S408, el motor de aprendizaje automático se entrena usando las características estáticas.

Después de entrenar el motor de aprendizaje automático usando la característica estática, el método puede incluir además las siguientes etapas 1 a 4.

En la etapa 1, en el caso de que se detecte un *software* sospechoso, se extrae un archivo ejecutable objetivo del *software* sospechoso.

En la etapa 2, el archivo ejecutable objetivo se introduce en el motor de aprendizaje automático entrenado para determinar, mediante el motor de aprendizaje automático entrenado, si el archivo ejecutable objetivo es un archivo ejecutable de un *software* malicioso.

En la etapa 3, si se determina mediante el motor de aprendizaje automático entrenado que el archivo ejecutable objetivo es el archivo ejecutable del *software* malicioso, se informa que la información de *software* del *software* sospechoso realiza la detección manual sobre el *software* sospechoso.

En la etapa 4, se recibe un resultado de detección manual, y se determina si la determinación actual mediante el motor de aprendizaje automático entrenado es errónea basándose en el resultado de detección manual. Si la determinación actual mediante el motor de aprendizaje automático entrenado es errónea, el motor de aprendizaje automático se entrena usando el archivo ejecutable objetivo como una nueva muestra de entrenamiento negativo.

En la realización, después de entrenar el motor de aprendizaje automático, el motor de aprendizaje automático entrenado se usa para detectar *software* malicioso. Dado que el motor de aprendizaje automático entrenado puede tener una precisión de detección baja, en la realización, si el motor de aprendizaje automático detecta un *software* malicioso, se informa del *software* malicioso y una detección manual (es decir, detección de nuevo) se realiza sobre el *software* malicioso. Si se determina basándose en la detección manual que el *software* sospechoso no es un *software* malicioso, se indica que la detección por el motor de aprendizaje automático es errónea. El motor de aprendizaje automático puede entrenarse nuevamente usando el archivo ejecutable objetivo del *software* sospechoso como una nueva muestra de entrenamiento negativo para mejorar la precisión de detección. Las muestras de entrenamiento incluyen muestras positivas y muestras negativas. Las muestras positivas corresponden a archivos ejecutables de *software* malicioso. Las muestras negativas corresponden a archivos ejecutables que no son de *software* malicioso*s*.

Se hace referencia a la figura 5, que es un diagrama estructural esquemático de un sistema para entrenar un motor de aprendizaje automático según una realización de la presente divulgación.

El sistema incluye un módulo de división de archivo 100, un módulo de extracción de características estáticas 200 y un módulo de entrenamiento 300.

El módulo de división de archivo 100 está configurado para obtener un archivo ejecutable y dividir el archivo ejecutable en múltiples partes a extraer basándose en un directorio de datos del archivo ejecutable.

El módulo de extracción de características estáticas 200 está configurado para extraer características de entropía de información de todas las partes a extraer y realizar la transformación de datos sobre las características de entropía de información para obtener características estáticas.

El módulo de entrenamiento 300 está configurado para entrenar un motor de aprendizaje automático usando las características estáticas.

En una realización, el módulo de división de archivo 100 incluye una unidad de obtención de archivos y una unidad de división.

La unidad de obtención de archivos está configurada para obtener el archivo ejecutable.

La unidad de división está configurada para dividir el archivo ejecutable en una parte de encabezado de archivo ejecutable, una parte de sección de archivo ejecutable, una parte de datos de superposición de archivo ejecutable, y una parte integral de archivo ejecutable basándose en el directorio de datos del archivo ejecutable.

En una realización, el módulo de extracción de características estáticas 200 incluye una unidad de extracción de características de entropía de información, una unidad de cálculo de flujo de entropía y una unidad de transformación de datos.

La unidad de extracción de características de entropía de información está configurada para extraer las características de entropía de información de todas las partes a extraer. Las características de entropía de información son datos binarios.

La unidad de cálculo de flujo de entropía está configurada para calcular una proporción de cada uno de los caracteres en los datos binarios correspondientes a las características de entropía de información con respecto a todos los caracteres en los datos binarios y obtener un flujo de entropía de las características de entropía de información basándose en la proporción.

La unidad de transformación de datos está configurada para realizar una transformada DCT o una transformada DWT sobre el flujo de entropía para obtener las características estáticas.

En una realización, la unidad de cálculo de flujo de entropía incluye una unidad secundaria de división, una unidad secundaria de cálculo de componente de flujo de entropía y una unidad de acumulación de componentes de flujo de entropía.

La unidad secundaria de división está configurada para dividir los datos binarios correspondientes a las características de entropía de información en múltiples bloques de bytes, donde cada uno de los bloques de bytes incluye 256 bytes.

La unidad secundaria de cálculo de componente de flujo de entropía está configurada para, para cada uno de los bloques de bytes, calcular una proporción de bloques de bytes p de cada uno de los caracteres en el bloque de bytes con respecto a todos los caracteres en el bloque de bytes, y calcular una componente de flujo de entropía S para cada uno de los bytes en el bloque de bytes usando la siguiente ecuación:

$$S = \left| p \cdot \log(p) \right|.$$

La unidad de acumulación de componentes de flujo de entropía está configurada para acumular todas las componentes de flujo de entropía para obtener el flujo de entropía.

En una realización, el sistema incluye además un módulo de determinación de estrategia de extracción.

El módulo de determinación de estrategia de extracción está configurado para determinar un formato de archivo del archivo ejecutable basándose en la información de formato general del archivo ejecutable, y seleccionar una estrategia de extracción de características basándose en el formato de archivo.

El módulo de extracción de características estáticas está configurado para extraer las características de entropía de información de todas las partes a extraer basándose en la estrategia de extracción de características, y realizar transformación de datos sobre las características de entropía de información para obtener las características estáticas.

En una realización, el módulo de determinación de estrategia de extracción incluye una unidad de determinación de formato de archivo y una unidad de selección de estrategia.

La unidad de determinación de formato de archivo está configurada para determinar valores de RVA de directorio de metadatos .Net y tamaño de directorio de metadatos .Net en el directorio de datos del archivo ejecutable, y determinar, si los valores de la RVA de directorio de metadatos .Net y el tamaño de directorio de metadatos .Net no están vacíos, que el formato de archivo del archivo ejecutable es formato .Net.

La unidad de selección de estrategia está configurada para seleccionar la estrategia de extracción de características basándose en el formato de archivo.

En una realización, el módulo de división de archivo está configurado para obtener el archivo ejecutable y dividir el archivo ejecutable en las múltiples partes a extraer basándose en una matriz de secciones del archivo ejecutable.

En una realización, el método incluye además un módulo de extracción y un módulo de determinación.

El módulo de extracción está configurado para extraer, en el caso de que se detecte un *software* sospechoso, un archivo ejecutable objetivo del *software* sospechoso.

El módulo de determinación está configurado para introducir el archivo ejecutable objetivo en el motor de aprendizaje automático entrenado para determinar mediante el motor de aprendizaje automático entrenado si el archivo ejecutable objetivo es un archivo ejecutable de un *software* malicioso.

En una realización, el método incluye además un módulo de informe y un módulo de entrenamiento secundario.

El módulo de informe está configurado para informar, si se determina mediante el motor de aprendizaje automático entrenado que el archivo ejecutable objetivo es el archivo ejecutable del *software* malicioso, de información de *software* del *software* sospechoso para realizar la detección manual sobre el *software* sospechoso.

El módulo de entrenamiento secundario está configurado para recibir un resultado de detección manual, determinar si la determinación actual mediante el motor de aprendizaje automático entrenado es errónea basándose en el resultado de detección manual, y entrenar, si la determinación actual mediante el motor de aprendizaje automático entrenado es errónea, el motor de aprendizaje automático usando el archivo ejecutable objetivo como una nueva muestra de entrenamiento negativo.

Dado que las realizaciones del sistema corresponden a las realizaciones del método, para las realizaciones del sistema, se pueden hacer referencias a la descripción de las realizaciones del método, que no se repiten en el presente documento.

Se proporciona además un medio de almacenamiento legible por ordenador que almacena un programa informático según la presente divulgación. El programa informático, cuando se ejecuta, realiza el método según las realizaciones anteriores. El medio de almacenamiento puede incluir medios capaces de almacenar códigos de programa, tal como un disco flash USB, un disco duro extraíble, una memoria de solo lectura (ROM), una memoria de acceso aleatorio (RAM), un disco magnético o un disco óptico.

Se proporciona además un dispositivo para entrenar un motor de aprendizaje automático según la presente divulgación. El dispositivo puede incluir una memoria y un procesador. La memoria almacena un programa informático. El procesador está configurado para ejecutar el programa informático almacenado en la memoria para realizar el método según las realizaciones anteriores. El dispositivo para entrenar un motor de aprendizaje automático puede incluir varias interfaces de red, una fuente de alimentación y otros componentes.

Las realizaciones en esta memoria descriptiva se describen de manera progresiva, cada una de las cuales enfatiza las diferencias con respecto a otras, y las partes iguales o similares entre las realizaciones pueden hacerse referencia unas con respecto a otras. Dado que el sistema dado a conocer en las realizaciones corresponde al método en el mismo, la descripción del mismo es relativamente simple, y para cuestiones relevantes se pueden hacer referencias a la descripción del método. Debe indicarse que para los expertos en la técnica, se pueden realizar muchas alteraciones y modificaciones sin apartarse del principio de la presente divulgación, y todas estas alteraciones y modificaciones caen dentro del alcance de protección de la presente divulgación.

Debe indicarse además que las terminologías de relación tales como “primero”, “segundo” en la presente descripción solo se usan en el presente documento para distinguir una entidad u operación de otra, en lugar de requerir o implicar que existe la relación u orden real entre las entidades u operaciones. Además, términos de “incluir”, “comprender” o

cualquier otra variante está destinada a ser no exclusiva. Por lo tanto, un proceso, un método, un artículo o un dispositivo que incluye múltiples componentes incluye no solo los componentes sino también otros componentes que no están enumerados, o también incluyen los componentes inherentes al proceso, el método, el artículo o el dispositivo. A menos que se limite de manera expresiva de otro modo, la declaración "que comprende (incluyendo) uno..." no excluye el caso de que puedan existir otros componentes similares en el proceso, el método, el artículo o el dispositivo.

REIVINDICACIONES

1. Un método para entrenar un motor de aprendizaje automático, que comprende:

obtener un archivo ejecutable, y dividir el archivo ejecutable en una pluralidad de partes a extraer basándose en un directorio de datos del archivo ejecutable;

extraer características de entropía de información de todas las partes a extraer, y realizar la transformación de datos sobre las características de entropía de información para obtener características estáticas; y

entrenar un motor de aprendizaje automático usando las características estáticas;

en el que la división del archivo ejecutable en una pluralidad de partes a extraer basándose en un directorio de datos del archivo ejecutable comprende:

dividir el archivo ejecutable en una parte de encabezado de archivo ejecutable, una parte de sección de archivo ejecutable, una parte de datos de superposición de archivo ejecutable, y una parte integral de archivo ejecutable basándose en el directorio de datos del archivo ejecutable.

2. El método según la reivindicación 1, en el que extraer características de entropía de información de todas las partes a extraer y realizar transformación de datos sobre las características de entropía de información para obtener características estáticas comprende:

extraer las características de entropía de información de todas las partes a extraer, en el que las características de entropía de información son datos binarios;

calcular una proporción de cada uno de los caracteres en los datos binarios correspondientes a las características de entropía de información con respecto a todos los caracteres en los datos binarios, y obtener un flujo de entropía de las características de entropía de información basándose en la proporción; y

realizar una transformada DCT o una transformada DWT sobre el flujo de entropía para obtener las características estáticas.

3. El método según la reivindicación 2, en el que el cálculo de una proporción de cada uno de los caracteres en los datos binarios correspondientes a las características de entropía de información con respecto a todos los caracteres en los datos binarios y la obtención de un flujo de entropía de las características de entropía de información basándose en la proporción comprende:

dividir los datos binarios correspondientes a las características de entropía de información en una pluralidad de bloques de bytes, en donde cada uno de los bloques de bytes comprende 256 bytes;

para cada uno de los bloques de bytes, calcular una proporción de bloques de bytes p de cada uno de los caracteres en el bloque de bytes con respecto a todos los caracteres en el bloque de bytes, y calcular una componente de flujo de entropía S para cada uno de los bytes en el bloque de bytes usando la siguiente ecuación:

$$S = \left| p \cdot \log(p) \right|;$$ y

acumular todas las componentes de flujo de entropía para obtener el flujo de entropía.

4. El método según la reivindicación 1, en el que después de obtener el archivo ejecutable, el método comprende además:

determinar un formato de archivo del archivo ejecutable basándose en información de formato general del archivo ejecutable, y seleccionar una estrategia de extracción de características basándose en el formato de archivo;

la extracción de características de entropía de información de todas las partes a extraer comprende:

extraer las características de entropía de información de todas las partes a extraer basándose en la estrategia de extracción de características.

5. El método según la reivindicación 4, en el que la determinación de un formato de archivo del archivo ejecutable basándose en información de formato general del archivo ejecutable comprende:

determinar valores de RVA de directorio de metadatos .Net y tamaño de directorio de metadatos .Net en el directorio de datos del archivo ejecutable; y

determinar, si los valores de la RVA de directorio de metadatos .Net y el tamaño de directorio de metadatos .Net no están vacíos, que el formato de archivo del archivo ejecutable es formato .Net.

6. El método según la reivindicación 1, en el que la división del archivo ejecutable en una pluralidad de partes a extraer basándose en un directorio de datos del archivo ejecutable comprende:

dividir el archivo ejecutable en una pluralidad de partes a extraer basándose en una matriz de secciones del archivo ejecutable.

7. El método según una cualquiera de las reivindicaciones 1 a 6, en el que después de entrenar un motor de aprendizaje automático usando las características estáticas, el método comprende además:

extraer, en el caso de que se detecte un *software* sospechoso, un archivo ejecutable objetivo del *software* sospechoso; e

introducir el archivo ejecutable objetivo en el motor de aprendizaje automático entrenado para determinar, mediante el motor de aprendizaje automático entrenado, si el archivo ejecutable objetivo es un archivo ejecutable de un *software* malicioso.

8. El método según la reivindicación 7, que comprende además:

informar, si se determina mediante el motor de aprendizaje automático entrenado que el archivo ejecutable objetivo es el archivo ejecutable del *software* malicioso, de información de *software* del *software* sospechoso para realizar la detección manual sobre el *software* sospechoso;

recibir un resultado de detección manual, y determinar si la determinación actual mediante el motor de aprendizaje automático entrenado es errónea basándose en el resultado de detección manual; y

entrenar, si la determinación actual mediante el motor de aprendizaje automático entrenado es errónea, el motor de aprendizaje automático usando el archivo ejecutable objetivo como una nueva muestra de entrenamiento negativo.

9. Un sistema para entrenar un motor de aprendizaje automático, que comprende:

un módulo de división de archivo, configurado para obtener un archivo ejecutable y dividir el archivo ejecutable en una pluralidad de partes a extraer basándose en un directorio de datos del archivo ejecutable;

un módulo de extracción de características estáticas, configurado para extraer características de entropía de información de todas las partes a extraer y realizar la transformación de datos sobre las características de entropía de información para obtener características estáticas; y

un módulo de entrenamiento, configurado para entrenar un motor de aprendizaje automático usando las características estáticas;

en el que el módulo de división de archivo comprende:

una unidad de obtención de archivos, configurada para obtener el archivo ejecutable; y

una unidad de división, configurada para dividir el archivo ejecutable en una parte de encabezado de archivo ejecutable, una parte de sección de archivo ejecutable, una parte de datos de superposición de archivo ejecutable, y una parte integral de archivo ejecutable basándose en el directorio de datos del archivo ejecutable.

10. El sistema según la reivindicación 9, en el que el módulo de extracción de características estáticas comprende:

una unidad de extracción de características de entropía de información, configurada para extraer las características de entropía de información de todas las partes a extraer, en el que las características de entropía de información son datos binarios;

una unidad de cálculo de flujo de entropía, configurada para calcular una proporción de cada uno de los caracteres en los datos binarios correspondientes a las características de entropía de información con

respecto a todos los caracteres en los datos binarios y obtener un flujo de entropía de las características de entropía de información basándose en la proporción; y

una unidad de transformación de datos, configurada para realizar una transformada DCT o una transformada DWT sobre el flujo de entropía para obtener las características estáticas.

11. El sistema según la reivindicación 9, que comprende además:

un módulo de determinación de estrategia de extracción, configurado para determinar un formato de archivo del archivo ejecutable basándose en información de formato general del archivo ejecutable, y seleccionar una estrategia de extracción de características basándose en el formato de archivo;

el módulo de extracción de características estáticas está configurado para extraer las características de entropía de información de todas las partes a extraer basándose en la estrategia de extracción de características, y realizar la transformación de datos sobre las características de entropía de información para obtener las características estáticas.

12. El sistema según una cualquiera de las reivindicaciones 9 a 11, que comprende además:

un módulo de extracción, configurado para extraer, en el caso de que se detecte un *software* sospechoso, un archivo ejecutable objetivo del *software* sospechoso; y

un módulo de determinación, configurado para introducir el archivo ejecutable objetivo en el motor de aprendizaje automático entrenado para determinar, mediante el motor de aprendizaje automático entrenado, si el archivo ejecutable objetivo es un archivo ejecutable de un *software* malicioso.

13. Un medio de almacenamiento legible por ordenador que almacena un programa informático, en el que el programa informático, cuando se ejecuta por un procesador, hace que el procesador realice el método para entrenar un motor de aprendizaje automático según una cualquiera de las reivindicaciones 1 a 8.

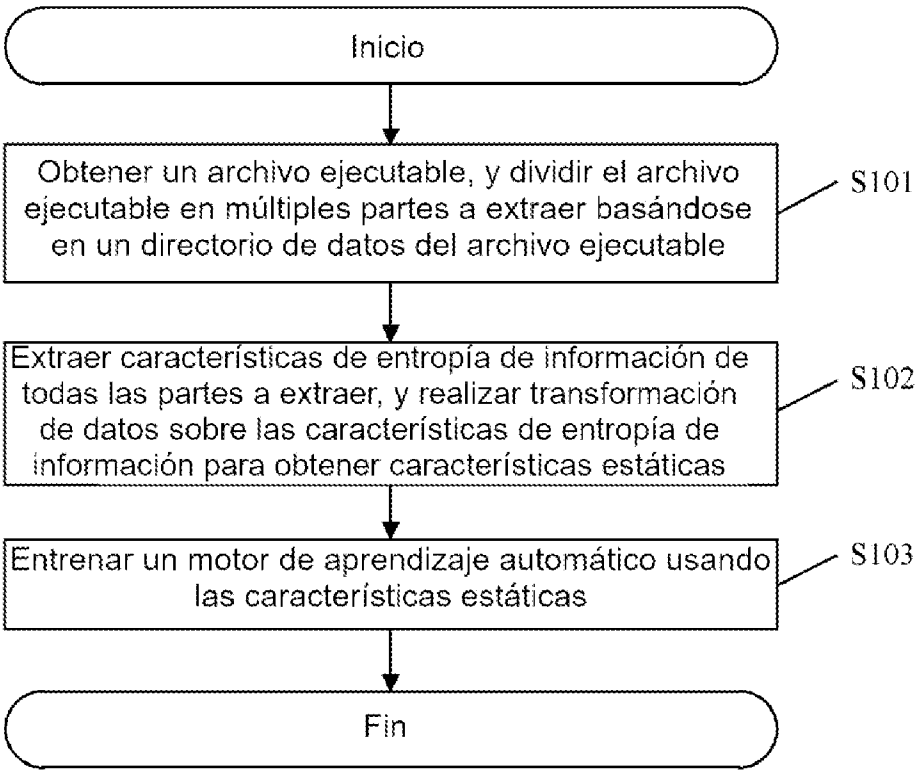

Figura 1

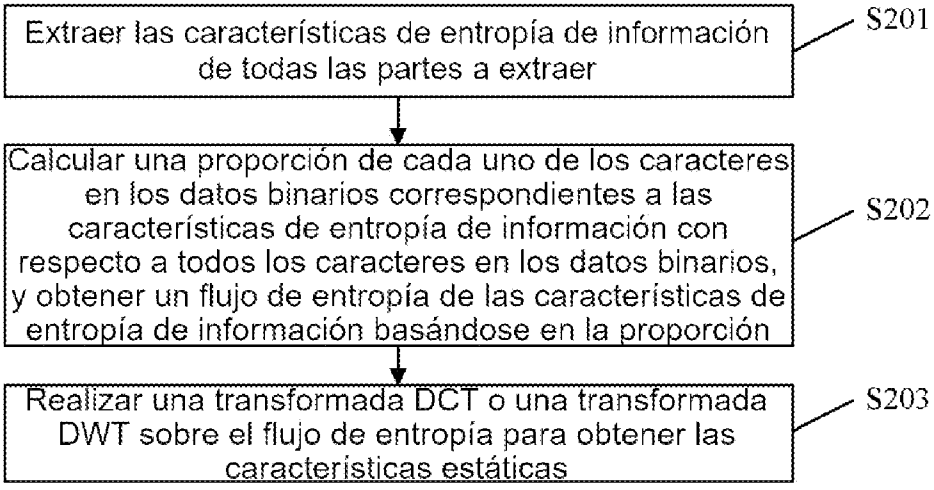

Figura 2

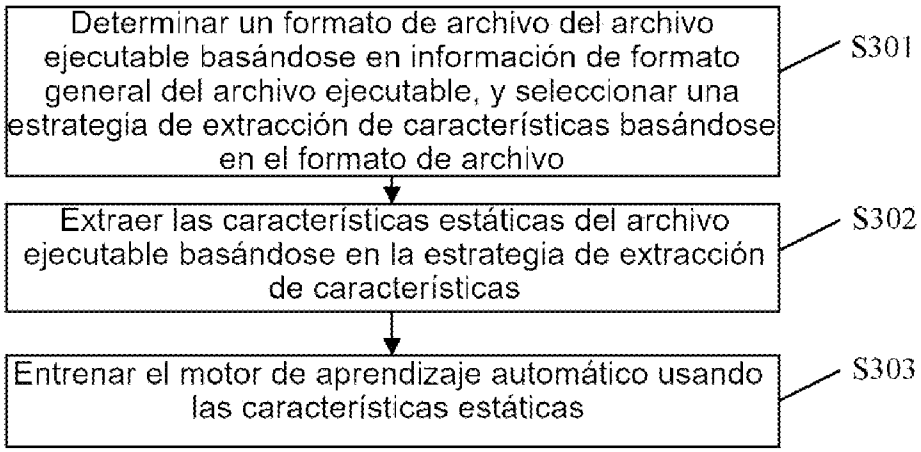
Determinar un formato de archivo del archivo ejecutable basándose en información de formato general del archivo ejecutable, y seleccionar una estrategia de extracción de características basándose en el formato de archivo
S301
Extraer las características estáticas del archivo ejecutable basándose en la estrategia de extracción de características
S302
Entrenar el motor de aprendizaje automático usando las características estáticas
S303

Figura 3

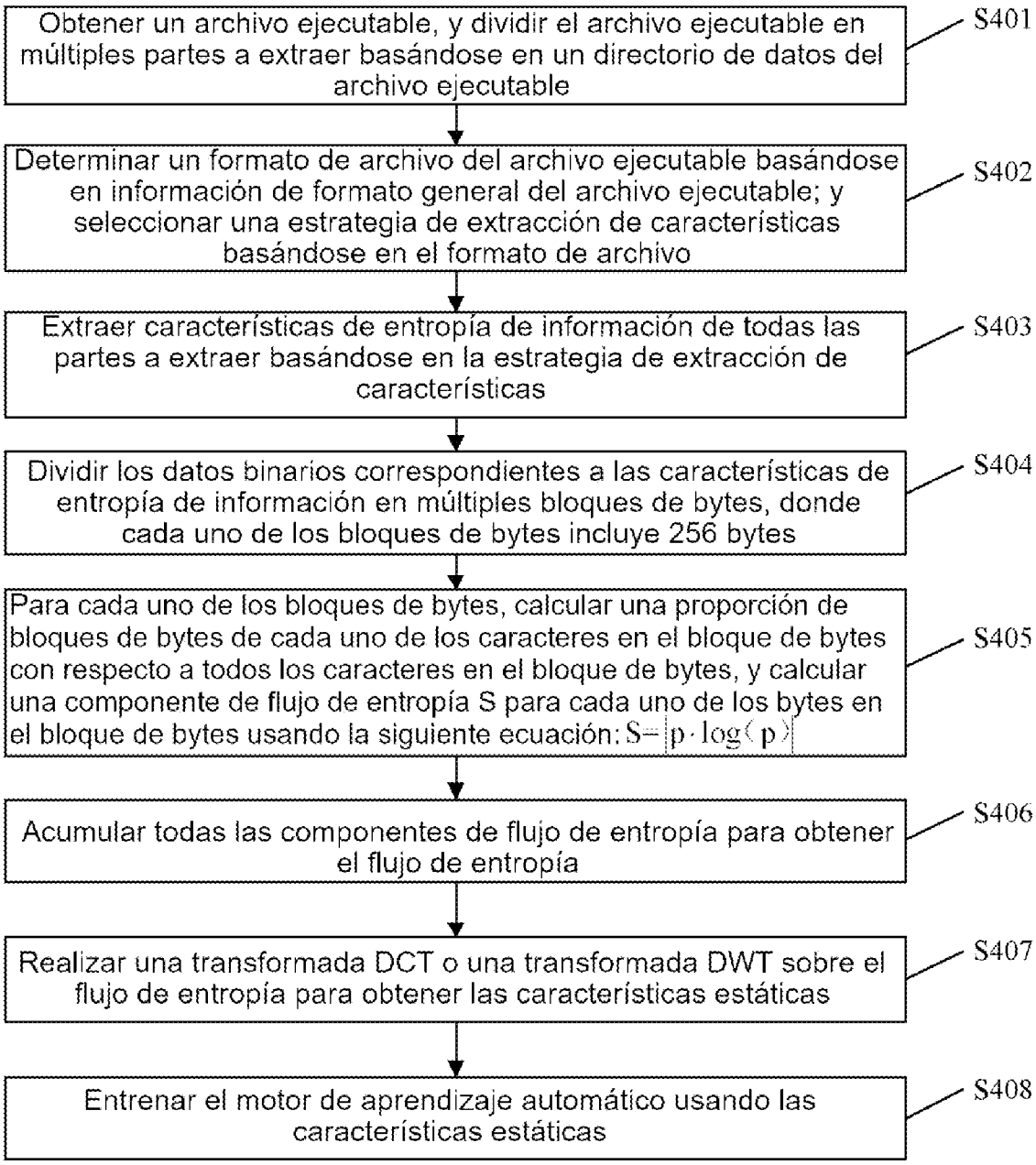

Figura 4

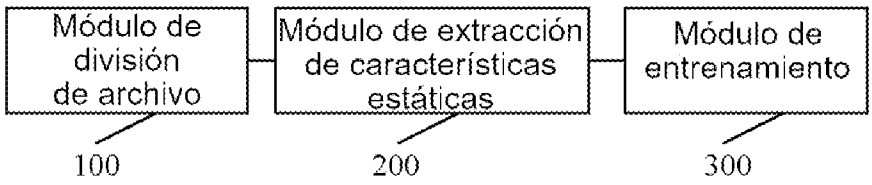

Figura 5