

**UN SISTEMA Y MÉTODO PARA IDENTIFICAR UNA VARIACIÓN DE UNA
FRASE EN UN PASAJE TEXTUAL**

REFERENCIA CRUZADA A SOLICITUDES RELACIONADAS Y

5

REIVINDICACIÓN DE PRIORIDAD

Esta solicitud reivindica prioridad a la
solicitud de patente de los EE. UU. núm. 18/378.509,
presentada el 10 de octubre de 2023, también titulada “A
10 System And Method Of Identifying A Variation Of A Phrase
In A Textual Passage”.

CAMPO DE LA INVENCION

15

La presente invención se relaciona generalmente
con el análisis automatizado de textos. Más
específicamente, la presente invención se relaciona con
identificar una variación de una frase en un pasaje
textual.

20

ANTECEDENTES DE LA INVENCION

Los representantes del centro de contacto
participan diariamente en grandes cantidades de
25 conversaciones con diversos clientes. Es posible que el
gerente o analista de un centro de contacto desee

HOJA DE SUSTITUCIÓN (REGLA 26)

analizar estas llamadas para obtener información para mejorar los servicios que brinda la compañía o la calidad del centro de contacto en sí. Para potenciar tales capacidades de procesos de análisis, un sistema de centro
5 de contacto puede exponer ciertas funcionalidades a un rol de “administrador” que puede ayudar al proceso de análisis, específicamente una funcionalidad útil para este, es encontrar un subconjunto de todas las conversaciones que traten un tema específico de interés.
10 Para encontrar automáticamente qué temas se incluyen en una conversación determinada, sería necesario describir cómo se definen dichos temas.

Un enfoque útil para definir tales temas es como un conjunto personalizable de frases típicas que pueden
15 aparecer en conversaciones que contengan el tema, donde una ‘frase’ puede ser una o más palabras consecutivas. Detectar una coincidencia exacta de la frase en el texto de la conversación es directo, sin embargo, compilar listas de todas las posibles variaciones de fraseo es
20 inmanejable. Por lo tanto, tal sistema debe dirigirse a generalizar efectivamente la identificación de variantes de frases semánticamente similares (p. ej., donde “consígueme a un gerente” debería considerarse similar a “transfiéreme a un supervisor”), sin requerir que el
25 usuario visualice de antemano todas las formas posibles de articulación.

Los sistemas de búsqueda actualmente disponibles comúnmente se basan en la coincidencia léxica para identificar la similitud semántica. Algunos enfoques tradicionales tratan las palabras como similares si son la misma palabra o se derivan de la misma forma lematizada o derivada.

Algunos enfoques pueden usar diversas técnicas de generalización, tal como aplicar diferentes pesos a diferentes palabras (p. ej., al calcular la frecuencia del término e inversa de la frecuencia del documento (TF/IDF, en inglés)), para evaluar la relevancia de las palabras dentro de un documento en relación con una colección de documentos (corpus). Tales sistemas de búsqueda pueden determinar que un tramo de texto constituye una ocurrencia de una frase buscada si hay suficientes palabras significativas comunes al texto examinado y una frase de interés.

Las variaciones de tales métodos tienden a sufrir de baja recuperación. Esto se debe a que (i) las palabras son comúnmente reemplazadas por sinónimos, y (ii) los conceptos pueden articularse de muchas maneras diversas. Por lo tanto, se siente la necesidad de encontrar variaciones semánticamente similares de frases.

Los enfoques basados en redes neuronales proporcionan métricas de distancia semántica, lo que permite al sistema cuantificar la similitud con palabras cercanas y relacionadas, p. ej., donde “gestor” está cerca de “supervisor”, pero introducen diferentes desafíos y limitaciones.

Los enfoques basados en redes neuronales (NN, en inglés) para detectar similitud de frases pueden dividirse en dos ramas:

10 Codificador cruzado: En tales implementaciones, 2 frases pueden concatenarse y alimentarse como entrada a un modelo NN preentrenado para clasificación binaria. Esta técnica es muy intensiva computacionalmente, ya que requiere un pase hacia adelante para cada par de frases
15 que necesitan determinarse para la similitud. Este enfoque no es adecuado para casos de uso tales como los de centros de llamadas, donde un usuario puede insertar muchas frases para buscar, y el número de oraciones en las conversaciones puede crecer constantemente,
20 resultando en un número demasiado alto de posibles combinaciones.

 Bicodificador: Este método representa todas las frases como un vector de incrustación del mismo tamaño producido por un modelo preentrenado. La similitud de
25 frases puede calcularse entonces como la distancia coseno entre los vectores. Esta técnica se diseña para frases

que son relativamente similares en longitud, y no se extiende trivialmente a la búsqueda dentro de un texto largo tal como conversaciones. Un enfoque ingenuo y popular incluiría separar el texto en oraciones o
5 unidades similares y después producir las incrustaciones, sin embargo esto no rinde resultados satisfactorios en el dominio de las conversaciones naturales, debido a sus características de oraciones largas y mal formadas.

Adicionalmente, los bicodificadores obtienen
10 incrustaciones de todas las palabras en la secuencia de entrada que resulta en la incapacidad de capturar matices de mayor resolución. En pocas palabras, la existencia de palabras irrelevantes en la secuencia de entrada exige umbrales más bajos para la similitud, que a su vez limita
15 la capacidad de capturar diferencias importantes.

Por ejemplo, una tal frase como “te llamé dos veces más temprano” es semánticamente muy diferente a “los llamaré más tarde” en el contexto de los centros de llamadas, sin embargo un umbral bajo puede impedir la
20 capacidad de detectar tales diferencias de manera eficiente incluso con la afinación de dominio.

Las modalidades de la invención pueden proporcionar alta, resolución a nivel de token de la representación semántica tal como la de un codificador
25 cruzado, sin el alto costo computacional que implica ejecutar un ‘pase hacia adelante’ para todas las

comparaciones por pares requeridas en la configuración del codificador cruzado.

Como se explica en la presente, mediante la producción de incrustaciones vectoriales para cada n-
5 grama o palabra, las modalidades de la invención pueden calcular la distancia para las incrustaciones vectoriales entre los términos de búsqueda y los n-gramas (p. ej., palabras) en el texto, eliminando de esta manera la necesidad de una cantidad cuádrlica de pases hacia
10 adelante.

Adicionalmente, las modalidades de la invención pueden encontrar efectivamente frases relevantes en representaciones textuales grandes, dinámicas, p. ej., que se originan a partir de conversaciones de los centros
15 de llamadas, sin la necesidad de definir los temas o frases por adelantado. Esto puede ser beneficioso, por ejemplo, al permitir que diferentes centros de contacto definan frases únicas de interés, y las actualicen a voluntad con un mayor grado de precisión que cualquier
20 método disponible actualmente.

Las modalidades de la invención pueden incluir un método para identificar la ocurrencia de una variación semántica de una frase en un pasaje por al menos un procesador.

25 De acuerdo con algunas modalidades, el al menos un procesador puede calcular un vector de incrustación de

frases, que representa un significado semántico de la frase basado en una representación textual de la frase; obtener una representación textual del pasaje que incluya una pluralidad de palabras; y extraer, de la
5 representación textual del pasaje, al menos un conjunto jerárquico de secuencias anidadas de palabras, donde cada secuencia de un conjunto jerárquico forma un subconjunto de palabras de una secuencia posterior en el conjunto.

El al menos un procesador puede calcular
10 posteriormente, para una o más (p. ej., cada) secuencia, un vector de incrustación de secuencias correspondiente, que representa un significado semántico de la secuencia. Para uno o más vectores de incrustación de secuencias, el al menos un procesador puede calcular un valor de
15 similitud vectorial correspondiente, que representa la similitud con el vector de incrustación de frases. El al menos un procesador puede entonces identificar una secuencia correspondiente a un valor máximo de similitud vectorial del uno o más valores de similitud vectorial, y
20 determinar la secuencia identificada como una variación semántica de la frase, basado en el valor máximo de similitud vectorial.

De acuerdo con algunas modalidades, el al menos un conjunto jerárquico puede incluir una pluralidad de
25 conjuntos jerárquicos. El al menos un procesador puede extraer la pluralidad de conjuntos jerárquicos al

seleccionar una pluralidad de secuencias núcleo, cada una incluye una o más palabras del pasaje, y para cada secuencia núcleo, producir un conjunto jerárquico respectivo de secuencias anidadas. Cada secuencia anidada
5 puede incluir la secuencia núcleo y las secuencias de palabras de anidación posteriores, que aumentan monótonamente.

De acuerdo con algunas modalidades, la representación textual del pasaje puede incluir una
10 transcripción de una conversación. El al menos un procesador puede seleccionar las secuencias núcleo al elegir una sección de la transcripción, asociada con un hablante específico en la conversación; y seleccionar las secuencias núcleo para comprender una o más palabras de
15 la sección elegida.

Adicional, o alternativamente, el al menos un procesador puede seleccionar las secuencias núcleo al calcular, para una o más palabras del pasaje, una o más etiquetas respectivas que representan partes del discurso
20 (POS, en inglés); y seleccionar las secuencias núcleo para comprender una o más palabras del pasaje basado en las etiquetas POS calculadas.

Adicional, o alternativamente, el al menos un procesador puede seleccionar las secuencias núcleo
25 calculando, para una o más palabras del pasaje, una o más métricas respectivas de relevancia del término; y

seleccionar las secuencias núcleo para comprender una o más palabras del pasaje basado en las métricas calculadas de relevancia del término.

De acuerdo con algunas modalidades, el al menos
5 un procesador puede calcular un vector de incrustación de secuencias de una secuencia específica mediante la obtención de un modelo basado en aprendizaje automático (ML, en inglés), preentrenado para mapear entre representaciones textuales de palabras y vectores de
10 incrustación de palabras correspondientes; inferir el modelo basado en ML en una o más palabras de la secuencia específica, para producir uno o más vectores de incrustación de palabras correspondientes, basado en dicho entrenamiento; y calcular el vector de incrustación
15 de secuencias como una función del uno o más vectores de incrustación de palabras.

El al menos un procesador puede entrenar el modelo ML al inferir el modelo basado en ML en una primera palabra, para producir un primer vector de
20 incrustación de palabras intermedias, que representa un significado semántico de la primera palabra; inferir el modelo basado en ML en una segunda palabra, para producir un segundo vector de incrustación de palabras provisionales, que representa un significado semántico de
25 la segunda palabra; recibir un primer elemento de datos de anotación, que representa la similitud semántica entre

la primera palabra y la segunda palabra; calcular un valor de similitud vectorial, que representa la similitud entre los vectores de incrustación de palabras provisionales; calcular un primer valor de función de pérdida, que representa una diferencia entre el valor de similitud vectorial y la similitud semántica, como se representa mediante el primer elemento de datos de anotación; y la capacitación del modelo ML para minimizar el primer valor de la función de pérdida.

Adicional, o alternativamente, el al menos un procesador puede entrenar el modelo ML al inferir el modelo basado en ML en una primera secuencia, para producir un primer vector de incrustación de secuencias intermedio; inferir el modelo basado en ML en una segunda secuencia, para producir un segundo vector de secuencia provisional; recibir un segundo elemento de datos de anotación, que representa la similitud semántica de secuencias entre la primera secuencia y la segunda secuencia; calcular un segundo valor de similitud vectorial, que representa la similitud entre los vectores de incrustación de secuencias provisionales; calcular un segundo valor de función de pérdida, que representa una diferencia entre el segundo valor de similitud vectorial y la similitud semántica de secuencias, como se representa mediante el segundo elemento de datos de

anotación; y entrenar el modelo ML para minimizar el segundo valor de la función de pérdida.

Adicional, o alternativamente, el al menos un procesador puede entrenar el modelo ML al recibir una
5 representación textual de una frase; recibir un elemento de datos de anotación de pasaje, que indica la existencia de una variación de la frase en el pasaje; inferir el modelo basado en ML en un conjunto jerárquico de secuencias obtenidas a partir de dicho pasaje, para
10 calcular un valor máximo de similitud vectorial provisional; y entrenar el modelo ML de tal manera que un valor del valor máximo de similitud vectorial provisional corresponda a la ocurrencia de una variación de la frase en el pasaje, como se representa mediante la anotación de
15 pasaje. Puede apreciarse que el elemento de datos de anotación de pasaje puede estar desprovisto de información que indique la ubicación de la variación de la frase dentro del pasaje recibido.

Las modalidades de la invención pueden incluir
20 un sistema para identificar la ocurrencia de una variación semántica de una frase en un pasaje. Las modalidades del sistema pueden incluir: un dispositivo de memoria no transitorio, en donde se almacenan módulos del código de instrucciones, y al menos un procesador
25 asociado con el dispositivo de memoria, y configurado para ejecutar los módulos del código de instrucciones.

Después de la ejecución de dichos módulos de código de instrucción, el al menos un procesador puede configurarse para: basado en una representación textual de la frase, calcular un vector de incrustación de frases, que representa un significado semántico de la frase; obtener una representación textual del pasaje, puede incluir una pluralidad de palabras; extraer, de la representación textual del pasaje, al menos un conjunto jerárquico de secuencias anidadas de palabras, en donde cada secuencia de un conjunto jerárquico forma un subconjunto de palabras de una secuencia posterior en el conjunto jerárquico; para cada secuencia, calcular un vector de incrustación de secuencias correspondiente, que representa un significado semántico de la secuencia; para uno o más vectores de incrustación de secuencias, calcular un valor de similitud vectorial correspondiente, que representa la similitud con el vector de incrustación de frases; identificar una secuencia correspondiente a un valor máximo de similitud vectorial del uno o más valores de similitud vectorial; y determinar la secuencia identificada como una variación semántica de la frase, basado en el valor máximo de similitud vectorial.

Las modalidades de la invención pueden incluir un método para identificar la ocurrencia de una variación semántica de una frase en un pasaje por al menos un procesador. Las modalidades del método pueden incluir:

basado en una representación textual de la frase,
calcular un vector de incrustación de frases, que
representa un significado semántico de la frase; recibir
una representación textual del pasaje, puede incluir una
5 pluralidad de n-gramos, extraer, a partir de la
representación textual del pasaje, una pluralidad de
secuencias de n-gramas; para cada secuencia, calcular un
vector de incrustación de secuencias correspondiente, que
representa un significado semántico de la secuencia; para
10 uno o más vectores de incrustación de secuencias,
calcular una similitud con el vector de incrustación de
frases; y determinar la secuencia identificada como una
variación semántica de la frase, basado en la similitud
calculada.

15

BREVE DESCRIPCIÓN DE LAS FIGURAS

La materia considerada como la invención se
señala particularmente y se reivindica claramente en la
20 parte final de la memoria descriptiva. Sin embargo, la
invención, tanto en cuanto a organización como método de
operación, junto con los objetos, características y
ventajas de esta, puede entenderse mejor haciendo
referencia a la siguiente descripción detallada cuando se
25 lea con las figuras adjuntas en las que:

La Figura 1 es un diagrama de bloques, que representa un dispositivo informático que puede incluirse en un sistema para identificar una variación de una frase en un pasaje textual de acuerdo con algunas modalidades
5 de la invención;

la Figura 2 es un diagrama de bloques, que representa un sistema para identificar una variación de una frase en un pasaje textual, de acuerdo con algunas modalidades de la invención;

10 la Figura 3 es un diagrama esquemático, que representa un proceso para calcular un vector de incrustación, de acuerdo con algunas modalidades de la invención;

la Figura 4 es un diagrama esquemático, que
15 representa un proceso de cálculo de vectores de incrustación de secuencias para conjuntos jerárquicos de secuencias anidadas de palabras, de acuerdo con algunas modalidades de la invención; y

la Figura 5 es un diagrama de flujo, que
20 representa un método para identificar una variación de una frase en un pasaje textual, de acuerdo con algunas modalidades de la invención.

Se entenderá que, por simplicidad y claridad de la ilustración, los elementos mostrados en las figuras no
25 se han dibujado necesariamente a escala. Por ejemplo, las dimensiones de algunos de los elementos pueden estar

exageradas con respecto a otros elementos para mayor claridad. Además, cuando se considere adecuado, los números de referencia pueden repetirse entre las figuras para indicar elementos correspondientes o análogos.

DESCRIPCIÓN DETALLADA DE LA PRESENTE INVENCION

Un experto en la técnica reconocerá que la invención puede incorporarse en otras formas específicas sin apartarse del espíritu o características esenciales de esta. Por lo tanto, las modalidades anteriores deben considerarse ilustrativas en todos los aspectos en lugar de limitar la invención descrita en la presente descripción. El alcance de la invención se indica, por lo tanto, mediante las reivindicaciones adjuntas, más que por la descripción anterior, y todos los cambios que entran dentro del significado y se pretende, por lo tanto, que el rango de equivalencia de las reivindicaciones se incluya en estas.

En la siguiente descripción detallada, se exponen numerosos detalles específicos para proporcionar una comprensión completa de la invención. Sin embargo, los expertos en la técnica entenderán que la presente invención puede practicarse sin estos detalles específicos. En otras instancias, los métodos, procedimientos y componentes muy conocidos no se han descrito en detalle para no oscurecer la presente invención. Algunas características o elementos descritos con respecto a una modalidad pueden combinarse con características o elementos descritos con respecto a otras modalidades. En aras de la claridad, puede que no

se repita un análisis de características o elementos iguales o similares.

Aunque las modalidades de la invención no se limitan en este sentido, los análisis que usan términos
5 tales como, por ejemplo, “que procesa”, “que computa”, “que calcula”, “que determina”, “que establece”, “que analiza”, “que verifica” o lo similar, pueden referirse a operaciones y/o procesos de una computadora, una plataforma informática, un sistema informático u otro
10 dispositivo electrónico informático, que manipula y/o transforma datos representados como cantidades físicas (p. ej., electrónicas) dentro de los registros y/o memorias de la computadora en otros datos representados de manera similar como cantidades físicas dentro de los
15 registros y/o memorias de la computadora u otro medio de almacenamiento no transitorio de información que puede almacenar instrucciones para realizar operaciones y/o procesos.

Aunque las modalidades de la invención no están
20 limitadas en este sentido, los términos “pluralidad” y “una pluralidad” como se usan en la presente descripción pueden incluir, por ejemplo, “múltiples” o “dos o más”. Los términos “pluralidad” o “una pluralidad” pueden usarse en toda la memoria descriptiva para describir dos o más
25 componentes, dispositivos, elementos, unidades,

parámetros o lo similar. El término “conjunto”, cuando se usa en la presente descripción, puede incluir uno o más elementos.

A menos que se indique explícitamente, las
5 modalidades del método descritas en la presente descripción no se limitan a un orden o secuencia particular. Adicionalmente, algunas de las modalidades del método descrito o elementos de este pueden ocurrir o realizarse simultáneamente, en el mismo punto en el
10 tiempo, o concurrentemente.

Ahora se hace referencia a la Figura 1, que es un diagrama de bloques que representa un dispositivo informático, que puede incluirse dentro de una modalidad de un sistema para identificar una variación de una frase
15 en un pasaje textual, de acuerdo con algunas modalidades.

El dispositivo informático 1 puede incluir un procesador o controlador 2 que puede ser, por ejemplo, un procesador de la unidad de procesamiento central (CPU, por sus siglas en inglés), un chip o cualquier
20 computadora adecuada o dispositivo computacional, un sistema operativo 3, una memoria 4, código ejecutable 5, un sistema de almacenamiento 6, dispositivos de entrada 7 y dispositivos de salida 8. El procesador 2 (o uno o más controladores o procesadores, posiblemente a través de
25 múltiples unidades o dispositivos) puede configurarse para realizar los métodos descritos en la presente

descripción, y/o ejecutar o actuar como los diversos
módulos, unidades, etc. Más de un dispositivo informático
1 puede incluirse en, y uno o más dispositivos
informáticos 1 pueden actuar como componentes de, un
5 sistema de acuerdo con las modalidades de la invención.

El sistema operativo 3 puede ser o puede incluir
cualquier segmento de código (p. ej., uno similar al
código ejecutable 5 descrito en la presente descripción)
diseñado y/o configurado para realizar tareas que
10 involucran la coordinación, planificación, arbitraje,
supervisión, control o de cualquier otra manera gestión
del funcionamiento del dispositivo informático 1, por
ejemplo, programación de programas o tareas del software
o permitir la comunicación de programas de software u
15 otros módulos o unidades. El sistema operativo 3 puede
ser un sistema operativo comercial. Se observará que un
sistema operativo 3 puede ser un componente opcional, por
ejemplo, en algunas modalidades, un sistema puede incluir
un dispositivo informático que no requiere o incluye un
20 sistema operativo 3.

La memoria 4 puede ser o puede incluir, por
ejemplo, una memoria de acceso aleatorio (RAM, por sus
siglas en inglés), una memoria de solo lectura (ROM, por
sus siglas en inglés), una RAM dinámica (DRAM, por sus
25 siglas en inglés), una DRAM sincrónica (SD-RAM), un chip
de memoria de tasa de datos doble (DDR, por sus siglas en

inglés), una memoria flash, una memoria volátil, una memoria no volátil, una memoria caché, un regulador, una unidad de memoria a corto plazo, una unidad de memoria a largo plazo, u otras unidades de memoria o unidades de almacenamiento adecuadas. La memoria 4 puede ser o puede incluir una pluralidad de unidades de memoria posiblemente diferentes. La memoria 4 puede ser un medio legible por computadora o procesador no transitorio, o un medio de almacenamiento no transitorio informático, por ejemplo, una RAM. En una modalidad, un medio de almacenamiento no transitorio tal como la memoria 4, una unidad de disco duro, otro dispositivo de almacenamiento, etc., puede almacenar instrucciones o código que cuando se ejecutan mediante un procesador puede hacer que el procesador realice métodos como se describe en la presente descripción.

El código ejecutable 5 puede ser cualquier código ejecutable, p. ej., una aplicación, un programa, un proceso, tarea o script. El código ejecutable 5 puede ejecutarse por el procesador o el controlador 2 posiblemente bajo el control del sistema operativo 3. Por ejemplo, el código ejecutable 5 puede ser una aplicación que puede identificar una variación de una frase en un pasaje textual como se describe más adelante en la presente. Aunque, en aras de la claridad, un único elemento del código ejecutable 5 se muestra en la Figura

1, un sistema de acuerdo con algunas modalidades de la invención puede incluir una pluralidad de segmentos de código ejecutable similares al código ejecutable 5 que puede cargarse en la memoria 4 y hacer que el procesador 2 realice los métodos descritos en la presente.

El sistema de almacenamiento 6 puede ser o puede incluir, por ejemplo, una memoria Flash como se conoce en la técnica, una memoria que es interna a, o incrustada en, un microcontrolador o chip como se conoce en la técnica, una unidad de disco duro, una unidad de accionamiento (CD-R, en inglés) que se extiende en CD, un disco Blu-ray (BD, en inglés), un dispositivo de bus serie universal (USB, en inglés) u otra unidad de almacenamiento extraíble y/o fija adecuada. Los datos textuales pueden almacenarse en el sistema de almacenamiento 6 y pueden cargarse desde el sistema de almacenamiento 6 en la memoria 4 donde puede procesarse mediante el procesador o el controlador 2. En algunas modalidades, algunos de los componentes mostrados en la Figura 1 pueden omitirse. Por ejemplo, la memoria 4 puede ser una memoria no volátil que tiene la capacidad de almacenamiento del sistema de almacenamiento 6. En consecuencia, aunque se muestra como un componente separado, el sistema de almacenamiento 6 puede estar incrustado o incluido en la memoria 4.

Los dispositivos de entrada 7 pueden ser o pueden incluir cualquier dispositivos de entrada, componentes, o sistemas adecuados, p. ej., un teclado o teclado numérico desmontable, un mouse y lo similar. Los
5 dispositivos de salida 8 pueden incluir uno o más pantallas o monitores, altavoces y/o cualquier otro dispositivo de salida adecuado (posiblemente desmontables). Cualquier dispositivo de entrada/salida (E/S) aplicable puede conectarse al dispositivo
10 informático 1 como se muestra por los bloques 7 y 8. Por ejemplo, una tarjeta de interfaz de red (NIC, en inglés) cableada o inalámbrica, un dispositivo de bus serie universal (USB) o un disco duro externo puede incluirse en los dispositivos de entrada 7 y/o dispositivos de
15 salida 8. Se reconocerá que cualquier número adecuado de dispositivos de entrada 7 y dispositivo de salida 8 puede conectarse operativamente al dispositivo informático 1 como se muestra en los bloques 7 y 8.

Un sistema de acuerdo con algunas modalidades de
20 la invención puede incluir componentes tales como, entre otros, una pluralidad de unidades centrales de procesamiento (CPU) o cualquier otro procesador o controlador de propósito múltiple adecuado o específico (p. ej., similar al elemento 2), una pluralidad de
25 unidades de entrada, una pluralidad de unidades de

salida, una pluralidad de unidades de memoria y una pluralidad de unidades de almacenamiento.

El término red neuronal (NN) o red neuronal artificial (ANN, en inglés), p. ej., una red neuronal que
5 implementa una función de aprendizaje automático (ML) o inteligencia artificial (AI, en inglés), puede usarse en la presente para referirse a un paradigma de procesamiento de información que puede incluir nodos, denominados neuronas, organizadas en capas, con enlaces
10 entre las neuronas. Los enlaces pueden transferir señales entre neuronas y pueden asociarse con ponderaciones. Una NN puede configurarse o entrenarse para una tarea específica, p. ej., reconocimiento o clasificación de patrones. El entrenamiento de la NN para la tarea
15 específica puede implicar afinar estas ponderaciones basado en ejemplos. Cada neurona de una capa intermedia o última puede recibir una señal de entrada, por ejemplo, una suma ponderada de señales de salida de otras neuronas, y puede procesar la señal de entrada usando una
20 función lineal o no lineal (p. ej., una función de activación). Los resultados de las capas de entrada e intermedias pueden transferirse a otras neuronas y los resultados de la capa de salida pueden proporcionarse como la salida de la NN. Típicamente, las neuronas y los
25 enlaces dentro de una NN se representan mediante constructos matemáticos, tales como funciones de

activación y matrices de elementos de datos y ponderaciones. Al menos un procesador (p. ej., el procesador 2 de la Figura 1) tal como una o más CPU o unidades de procesamiento de gráficos (GPU, en inglés), o
5 un dispositivo de hardware dedicado puede realizar los cálculos relevantes.

Con referencia ahora a la Figura 2, que representa un sistema 10 para identificar una variación de una frase en un pasaje textual, de acuerdo con algunas
10 modalidades.

De acuerdo con algunas modalidades de la invención, el sistema 10 puede implementarse como un módulo de software, un módulo de hardware, o cualquier combinación de estos. Por ejemplo, el sistema puede ser o
15 puede incluir un dispositivo informático tal como el elemento 1 de la Figura 1, y puede adaptarse para ejecutar uno o más módulos del código ejecutable (p. ej., elemento 5 de la Figura 1) para identificar una variación de una frase en un pasaje textual, como se describe más
20 adelante en la presente.

Como se muestra en la Figura 2, las flechas pueden representar el flujo de uno o más elementos de datos hacia y desde el sistema 10 y/o entre los módulos o elementos del sistema 10. Algunas flechas se han omitido
25 en la Figura 2 para fines de claridad.

Como se muestra en el ejemplo de la Figura 2, el sistema 10 puede incluir un módulo de preprocesamiento 200 y un módulo de análisis 100.

El módulo de preprocesamiento 200 puede configurarse para obtener información textual, tal como una representación textual de un pasaje (“texto de pasaje 25PT”) y una representación textual de una frase (“texto de frase 27PH”). para analizarse para identificar una variación de una frase de interés en un pasaje textual. El módulo de análisis 100 puede recibir información textual 25PT/27PH, del módulo de preprocesamiento 200 y puede a su vez analizar información textual 25PT/27PH para identificar, o determinar la aparición de una frase de interés en el pasaje, como se elabora en la presente.

De acuerdo con algunas modalidades, el módulo de preprocesamiento 200 puede recibir al menos uno de texto de pasaje 25PT y texto de frase 27PH como entrada, p. ej., a través del dispositivo de entrada 7 de la Figura 1.

Adicional, o alternativamente, el módulo de preprocesamiento 200 puede recibir información de una pluralidad de fuentes, en una variedad de formatos, y aplicar cualquier lógica para procesar esta información en representación textual (p. ej., texto de pasaje 25PT y texto de frase 27PH).

Por ejemplo, el módulo de preprocesamiento 200 puede recibir datos de audio 20A (p. ej., un archivo de audio, o datos de audio en flujo continuo) que incluyen voz humana, p. ej., a través del dispositivo de entrada 7 de la Figura 1, y/o desde el almacenamiento 6 de la Figura 1. El módulo de preprocesamiento 200 puede usar un proceso o módulo de voz a texto 220 en los datos de audio entrantes 20A, como se conoce en la técnica, para obtener el texto de pasaje 25PT y/o el texto de frase 27PH como una representación textual de la voz en los datos de audio 20A.

Como se conoce en la técnica, los vectores de incrustación de palabras son representaciones numéricas de palabras en un espacio vectorial de alta dimensión. Esta representación está diseñada para captar el significado semántico y las relaciones entre palabras basado en su contexto en un corpus de texto. Las incrustaciones de palabras se han convertido en una herramienta fundamental en las técnicas de procesamiento del lenguaje natural (PNL) y aprendizaje automático (ML) porque permiten que los algoritmos funcionen con palabras como vectores continuos en lugar de símbolos discretos. Las palabras de un vocabulario típicamente se asignan a vectores numéricos de incrustación de palabras, elegidos de manera que codifiquen las relaciones semánticas entre las palabras, donde las palabras similares se representan

como vectores que están próximos entre sí en el espacio vectorial.

Como se muestra en la Figura 2, el sistema 10 puede incluir un modelo de incrustación de palabras basado en ML 110 (o “modelo 110” para abreviar), configurado para mapear entre una representación textual de una palabra, y un vector de incrustación de palabras correspondiente 110WE. La estructura y configuración del modelo 110 se elabora en la presente descripción, p. ej., en relación con las Figuras 3 y 4.

Como se muestra en la Figura 2, el sistema 10 puede incluir además un módulo de incrustación de secuencias 140, configurado para calcular un vector de incrustación de una oración, o secuencia de palabras, basado en, o como una función de los vectores de incrustación de palabras 110WE de palabras que comprenden esa secuencia.

Por ejemplo, el módulo de incrustación de secuencias 140 puede recibir (i) la representación textual, texto de frases 27PH, de una frase de interés, y (ii) vectores de incrustación de palabras 110WE de palabras 27W en texto de frases 27PH, y calcular un vector de incrustación de frases 140PE que representa un significado semántico de la frase, basado en (i) y (ii). Por ejemplo, el vector de incrustación de frases 140PE puede ser un vector promedio, donde cada entrada del

vector es un promedio de las entradas respectivas en los vectores de incrustación de palabras 110WE de una o más (p. ej., todas las) palabras 27W en el texto de frase 27PH.

5 Como se muestra en la Figura 2, el sistema 10 puede incluir un módulo de generación de conjuntos jerárquicos 120 (o un módulo jerárquico 120, para abreviar). De acuerdo con algunas modalidades, el módulo jerárquico 120 puede configurarse para obtener una
10 representación textual 25PT de un pasaje de interés compuesto por una pluralidad de palabras 25W, y extraer de la representación textual 25PT al menos un conjunto jerárquico 120HS de secuencias anidadas 25SQ de palabras 25W.

15 El término “anidado” puede usarse en la presente en el sentido de que (a) cada conjunto jerárquico 120HS puede originarse a partir de una secuencia núcleo 25KR de una o más palabras 25W, y (b) cada secuencia 25SQ de un conjunto jerárquico puede formar un subconjunto de
20 palabras 25W de una secuencia posterior en el conjunto jerárquico 120HS.

 Por ejemplo, una representación textual 25PT de la oración “Me gustan las pizzas frías” puede usarse para producir un primer conjunto jerárquico 120HS de
25 secuencias 25SQ tales como: {“Me”; “Me gusta”; “Me gusta

frío”; “Me gustan las pizzas frías”}, donde “Me” es la secuencia núcleo 25KR.

En otro ejemplo, puede usarse “pizzas frías” como una secuencia núcleo 25KR, para producir un segundo
5 conjunto jerárquico 120HS de secuencias 25SQ tales como:
{“pizzas frías”; “como pizzas frías”; “Me gustan las pizzas frías”}.

De acuerdo con algunas modalidades, el módulo de generación de conjuntos jerárquicos 120 puede producir
10 una pluralidad de conjuntos jerárquicos 120HS.

Por ejemplo, el módulo de generación de conjuntos jerárquicos 120 puede configurarse con un tamaño de secuencia mínimo, que define un número mínimo de palabras 25W o n-gramas en una secuencia 25SQ (p. ej.,
15 ‘1’), y un tamaño máximo de secuencia, que define un número máximo de palabras 25W o n-gramas en una secuencia 25SQ (p. ej., igual a la longitud del texto de pasaje 25PT). El módulo de generación de conjuntos jerárquicos 120 puede producir una combinación exhaustiva de
20 conjuntos jerárquicos, que pueden incluir todas las combinaciones posibles de secuencias anidadas 25SQ entre el tamaño de secuencia mínimo y el tamaño de secuencia máximo.

Adicional, o alternativamente, el módulo de
25 generación de conjuntos jerárquicos 120 puede seleccionar

una pluralidad de secuencias núcleo 25KR, cada una incluye una o más palabras 25W (o tokens, n-gramas, etc.) 25W del pasaje 25PT, p. ej., como se define mediante el número mínimo de palabras 25W o n-gramas en una secuencia 5 25SQ. Para una o más (p. ej., cada) secuencia núcleo, el módulo de generación de conjuntos jerárquicos 120 puede producir un conjunto jerárquico 120HS respectivo de secuencias anidadas. Cada conjunto jerárquico 120HS puede incluir la secuencia núcleo respectiva, y secuencias 25SQ 10 de palabras 25W de anidación posteriores que aumentan monótonamente.

De acuerdo con algunas modalidades, los datos de audio 20A pueden incluir una conversación entre uno o más hablantes, tal como una conversación entre un agente y un 15 cliente en un centro de llamadas. El módulo de voz a texto 220 puede producir una representación textual 25PT que es una transcripción de la conversación. Adicionalmente, el módulo de voz a texto 220 puede colaborar con un módulo de reconocimiento de hablantes, 20 adaptado para asociar porciones de la conversación con uno o más hablantes específicos, tal como se conoce en la técnica.

El sistema 10 puede seleccionar una sección de la transcripción que se asocia con un hablante de interés 25 específico en la conversación, y puede limitar la

selección de al menos una secuencia núcleo 25KR a la sección seleccionada de ese hablante.

En otras palabras, para minimizar el intervalo de contenido que puede buscarse dentro del texto de pasaje 25PT, el módulo de preprocesamiento 200 puede dividir la conversación en diferentes secciones de búsqueda, marcando las divisiones por indicaciones de cambio de hablante y/o lapso de tiempo de silencio mínimo (p. ej., 1 segundo). El sistema 10 puede entonces diferenciar entre enunciados de diferentes hablantes (p. ej., agentes y clientes), y limitar la búsqueda de variaciones de la frase 27PH a secciones de un hablante de interés. El sistema 10 puede de esta manera eliminar la necesidad de buscar sobre texto que se extiende o se superpone a más de una sección (p. ej., hablante) en un texto de pasaje 25PT.

En tales modalidades, el módulo de generación de conjuntos jerárquicos 120 puede seleccionar secuencias núcleo 25KR para incluir una o más palabras 25W de la sección seleccionada, p. ej., del hablante de interés. Adicional, o alternativamente, el módulo de generación de conjuntos jerárquicos 120 puede producir el conjunto de conjuntos jerárquicos, anidados 120HS para incluir únicamente secuencias de palabras 25W de la sección seleccionada, p. ej., del hablante de interés.

De acuerdo con algunas modalidades, para una o más (p. ej., cada) secuencia 25SQ de los conjuntos jerárquicos 120HS, el módulo de incrustación de secuencias 140 puede calcular un vector de incrustación de secuencias 140SE correspondiente, que representa un significado semántico de la secuencia, basado en las palabras 25W de los vectores de incrustación 110WE en esa secuencia 25SQ.

Por ejemplo, el vector de incrustación de secuencia 140SE puede ser un vector promedio, donde cada entrada del vector es un promedio de las entradas respectivas en los vectores de incrustación de palabras 110WE de la secuencia 25SQ.

Como se muestra en la Figura 2, el sistema puede incluir un módulo de cálculo de similitud configurado para calcular la similitud entre vectores de incrustación, basado en una métrica de distancia predefinida.

De acuerdo con algunas modalidades, para uno o más (p. ej., todos los) vectores de incrustación de secuencias 140SE, el módulo de cálculo de similitud puede calcular un valor de similitud vectorial correspondiente, que representa una similitud respectiva con el vector de incrustación de frases.

Por ejemplo, la métrica de distancia puede incluir una métrica de distancia coseno, y el valor de

similitud 150S puede ser un valor numérico que define la similitud de uno o más (p. ej., todos los) vectores de incrustación de secuencias 140SE para incrustar el vector de incrustación de frases 140PE. En tales modalidades, un
5 valor de alta similitud 150S puede representar una alta similitud semántica entre una frase específica 27PH (representada por un vector de incrustación de frases 140PE) y una secuencia específica 25SQ de palabras 25W (representada por un vector de incrustación de secuencias
10 específico 140SE).

El módulo de cálculo de similitud 150 puede identificar posteriormente una secuencia específica 25SQ que corresponde a un valor máximo de similitud vectorial 150S del uno o más valores de similitud vectorial. En
15 otras palabras, el módulo de cálculo de similitud 150 puede encontrar la secuencia 25SQ que es más semánticamente similar a la frase 27PH de interés.

El sistema 10 puede determinar si la secuencia identificada 25SQ es una variación semántica de la frase
20 27PH, basado en el valor máximo de similitud vectorial.

Por ejemplo, cuando el valor máximo de similitud vectorial 150S, correspondiente a una secuencia específica 25SQ de palabras 25W, supera un umbral predefinido, entonces el sistema 10 puede producir un
25 elemento de datos de indicación 150IN. El elemento de datos de indicación 150IN puede incluir una notificación

que identifica la secuencia específica 25SQ de palabras 25W (que se incluye en el texto de pasaje 25PT) como idéntica a, o como una variación semántica de frase de interés 27PH.

5 Ahora se hace referencia a la Figura 3, que es un diagrama de bloques, que representa un proceso para calcular un vector de incrustación, de acuerdo con algunas modalidades de la invención. Los módulos representados en la Figura 3 pueden ser los mismos que
10 los módulos numerados respectivamente de la Figura 2.

 También se hace referencia a la Figura 4, que es un diagrama esquemático, que representa un proceso para calcular vectores de incrustación de secuencias para conjuntos jerárquicos de secuencias anidadas de palabras
15 25W, de acuerdo con algunas modalidades de la invención. Los módulos representados en la Figura 4 pueden ser los mismos que los módulos numerados respectivamente de las Figuras 2 y 3.

 Como se muestra en el ejemplo de la Figura 3, el
20 modelo de incrustación de palabras 110 puede ser, o puede incluir un modelo basado en ML 110, tal como un modelo NN 110. El modelo ML 110 puede preentrenarse para mapear entre representaciones textuales de palabras 25W/27W, tales como texto de frase 27PH y vectores de incrustación
25 de palabras correspondientes, de acuerdo con la ecuación Eq. 1, a continuación:

$$W(< w_1, w_2, \dots, w_m >) = < \vec{v}_1, \vec{v}_2, \dots, \vec{v}_m >$$

donde $< w_1, w_2, \dots, w_m >$ es una secuencia de entrada
(25SQ/ 27PH) de ‘m’ palabras o tokens,
5 $< \vec{v}_1, \vec{v}_2, \dots, \vec{v}_m >$ es un conjunto correspondiente de
incrustaciones de palabras ‘m’ 110WE, y
W es una función basada en ML 110 diferenciable
que puede producir incrustaciones
10 contextualizadas 110WE para cada token w_i en la
secuencia de entrada (25SQ/27PH).

De acuerdo con algunas modalidades, y como se
muestra en la Figura 3, el sistema 10 puede inferir el
modelo basado en ML 110 en una o más palabras 27W
15 $< w_1, w_2, \dots, w_m >$ de un texto de frase específico 27PH. El
modelo basado en ML 110 puede producir por lo tanto uno o
más vectores de incrustación de palabras correspondientes
110WE $< \vec{v}_1, \vec{v}_2, \dots, \vec{v}_m >$, basados en el entrenamiento.

El módulo de incrustación de secuencias 140
20 puede calcular posteriormente un vector de incrustación
de frases 140PE como una función (denotada como $F(\cdot)$, p.
ej., un promedio puntual) del uno o más vectores de
incrustación de palabras 110WE $< \vec{v}_1, \vec{v}_2, \dots, \vec{v}_m >$.

De acuerdo con algunas modalidades, y como se muestra en la Figura 4, el sistema 10 puede inferir el modelo basado en ML 110 en una o más palabras 25W $\langle w_1, w_2, \dots, w_n \rangle$ de al menos una secuencia 25SQ. El modelo
5 basado en ML 110 puede producir por lo tanto uno o más vectores de incrustación de palabras correspondientes 110WE $\langle \vec{u}_1, \vec{u}_2, \dots, \vec{u}_n \rangle$, basados en el entrenamiento.

Como se conoce en el arte del procesamiento del lenguaje natural (PNL), las etiquetas de las partes del
10 discurso (POS) son etiquetas asignadas a palabras individuales o tokens en un texto para indicar su categoría gramatical o función sintáctica dentro de una oración. Esto incluye marcadores sintácticos tales como verbos, adjetivos, adverbios, pronombres, preposiciones,
15 conjunciones, y lo similar.

De acuerdo con algunas modalidades, el módulo de generación de conjuntos jerárquicos 120 puede calcular, para una o más palabras 25W o tokens en texto de pasaje 25PT, etiquetas POS correspondientes 120POS, como se
20 conoce en la técnica.

El módulo de generación de conjuntos jerárquicos 120 puede seleccionar posteriormente secuencias núcleo 25KR, cada una incluye una o más palabras 25W del pasaje 25PT basado en las etiquetas 120POS calculadas. Por
25 ejemplo, el módulo de generación de conjuntos jerárquicos 120 puede seleccionar secuencias núcleo 25KR que

HOJA DE SUSTITUCIÓN (REGLA 26)

incluyen, o representan roles semánticos predeterminados o POS, tales como frases verbales, como se indica mediante las etiquetas POS 120POS.

Adicional, o alternativamente, el módulo de
5 generación de conjuntos jerárquicos 120 puede seleccionar secuencias núcleo 25KR que pertenecen a un hablante específico, una sección específica en el texto 25PT, u otras métricas de relevancia terminológica, tal como TF/IDF, de manera que las secuencias núcleo 25KR se
10 seleccionan como las palabras más relevantes 25W, o combinación de palabras 25W en el texto 25PT.

En el ejemplo no limitante de la Figura 4, las secuencias núcleo seleccionadas 25KR son W1 y W2, que corresponden a vectores de incrustación de palabras
15 $\vec{u}_1, \vec{u}_2$ respectivamente.

Como se elabora en la presente, el módulo de generación de conjuntos jerárquicos 120 puede usar las secuencias núcleo seleccionadas 25KR para producir el conjunto jerárquico 120HS respectivo de secuencias
20 anidadas.

La Figura 4 proporciona ejemplos de incrustaciones de palabras 110WE de dos conjuntos jerárquicos 120HS: Uno se origina a partir de $\vec{u}_1$ (incrustación de W₁) y culmina en $\vec{u}_1, \vec{u}_2, \dots, \vec{u}_n$. Otro se

origina a partir de $\vec{u}_2$ (incrustación de W_2) y también culmina en $\vec{u}_1, \vec{u}_2, \dots, \vec{u}_n$.

Como se muestra en la Figura 4, el módulo de incrustación de secuencias 140 puede producir vectores de incrustación de secuencias 140SE para una o más (p. ej.,
5 cada) secuencia 25SQ en uno o más (p. ej., cada) conjunto jerárquico 120HS.

El módulo de incrustación de secuencias 140 puede calcular posteriormente el (los) vector(es) de incrustación de secuencias 140SE como una función
10 (denotada como $F(\cdot)$, p. ej., un promedio puntual) del uno o más vectores $\langle \vec{u}_1, \vec{u}_2, \dots, \vec{u}_n \rangle$ de incrustación de palabras de cada secuencia de palabras 25SQ.

De acuerdo con algunas modalidades, el sistema
15 10 puede incluir un módulo de entrenamiento 120, adaptado para entrenar el modelo de incrustación de palabras basado en ML 110 (también denominado en la presente modelo NN 110) para mapear entre palabras (p. ej., 25W, 27W) y vectores de incrustación de palabras 110WE
20 correspondientes.

Puede apreciarse que el módulo de entrenamiento 120 puede entrenar el modelo ML 110 en una etapa de entrenamiento inicial, donde los pesos de un modelo NN 110 pueden adaptarse basado un conjunto de datos de
25 entrenamiento 29AN anotado. La etapa de entrenamiento puede seguirse mediante una etapa de inferencia, donde el

modelo basado en ML 110 puede aplicarse a, o inferirse en la frase objetivo 27PH y/o texto de pasaje 25PT de interés.

Adicional, o alternativamente, el módulo de
5 entrenamiento 120 puede entrenar el modelo de ML 110 repetidamente (p. ej., continuamente, a lo largo del tiempo) a medida que se presentan nuevos ejemplos anotados de datos anotados al sistema 10.

De acuerdo con algunas modalidades, el módulo de
10 entrenamiento 120 puede entrenar el modelo de ML 110 incrementalmente, basado en valores calculados de una o más funciones de pérdida 120LF, como se elabora en la presente.

Por ejemplo, el sistema 10 puede inferir el
15 modelo basado en ML 110 en una primera palabra (p. ej., incluido en el texto de pasaje 25PT), para producir un primer vector de incrustación de palabras provisionales 110WE, que representa un significado semántico de la primera palabra. El sistema 10 también puede inferir el
20 modelo basado en ML 110 en una segunda palabra (p. ej., también incluido en el texto de pasaje 25PT), para producir un segundo vector de incrustación de palabras provisionales 110WE que representa un significado semántico de la segunda palabra.

25 El término “provisional” puede usarse en la presente en el sentido de que durante el entrenamiento,

los vectores de incrustación 110WE pueden tener valores temporales, inexactos que pueden adaptarse o refinarse gradualmente como parte del proceso de entrenamiento para representar mejor las palabras de interés subyacentes.

5 De acuerdo con algunas modalidades, el módulo de entrenamiento 120 puede recibir después (p. ej., a través del dispositivo de entrada 7 de la Figura 1) un primer elemento de datos de anotación 29AN. El elemento de datos de anotación 29AN puede representar, o indicar similitud
10 semántica entre la primera palabra y la segunda palabra. Por ejemplo, el elemento de datos de anotación 29AN puede ser un valor numérico entre $[0, 1]$, donde un valor bajo (p. ej., 0,1) representa baja similitud semántica entre la primera palabra y la segunda palabra (p. ej., “perro”,
15 “sombrero”), y un valor alto (p. ej., 0,9) representa una alta similitud semántica entre la primera palabra y la segunda palabra (p. ej., “perro”, “sabueso”).

De acuerdo con algunas modalidades, el módulo de similitud 150 puede calcular un valor de similitud
20 vectorial 150S (p. ej., una inversa de la distancia coseno) que puede representar similitud entre el vector de incrustación de palabras provisionales 110WE de la primera palabra y el vector de incrustación de palabras provisionales 110WE de la segunda palabra.

25 El módulo de entrenamiento 120 puede calcular un primer valor de función de pérdida numérica 120LF

(denotado 120LF1), que representa una diferencia entre el valor de similitud vectorial 150S y la similitud semántica, como se representa mediante el primer elemento de datos de anotación 29AN.

5 Por ejemplo, el valor de función de pérdida 120LF (120LF1) puede estar en el rango de $[0, 1]$, donde un valor numérico grande (p. ej., 0,9) representa una gran diferencia entre el valor de similitud vectorial 150S y el elemento de datos de anotación 29AN, y donde un
10 valor numérico pequeño (p. ej., 0,1) representa una pequeña diferencia entre el valor de similitud vectorial 150S y el elemento de datos de anotación 29AN.

El módulo de entrenamiento 120 puede entonces entrenar el modelo de ML 110 para minimizar el valor de
15 la función de pérdida 120LF (120LF1).

En otras palabras, el módulo de entrenamiento 120 puede utilizar el elemento de datos de anotación 29AN como información de supervisión para entrenar el modelo de ML 110. Por ejemplo, el módulo de entrenamiento 120
20 puede entrenar el modelo ML 110 al aplicar un algoritmo de retropropagación como se conoce en la técnica, para ajustar pesos de NN 110 disminuyendo de esta manera la diferencia entre el valor de similitud 150S y el elemento de datos de anotación 29AN, y minimizando el valor de
25 función de pérdida 120LF (120LF1).

Adicional, o alternativamente, el módulo de entrenamiento 120 puede entrenar el modelo ML 110 basado en vectores de incrustación de secuencias.

Por ejemplo, el módulo de entrenamiento 120 puede inferir el modelo basado en ML 110 en una primera secuencia 25SQ (p. ej., del texto de pasaje 25PT), para producir un primer vector de incrustación de secuencias provisional 140SE, e inferir el modelo basado en ML 110 en una segunda secuencia 25SQ para producir un segundo vector de secuencia provisional 140SE. El módulo de entrenamiento 120 puede recibir un segundo elemento de datos de anotación 29AN que representa la similitud semántica de secuencias entre la primera secuencia 25SQ y la segunda secuencia 25SQ. Por ejemplo, el elemento de datos de anotación 29AN puede ser un valor numérico entre [0, 1], donde un valor bajo (p. ej., 0,1) representa baja similitud semántica entre la primera secuencia 25SQ y la segunda secuencia 25SQ (p. ej., “Me gusta la torta”, “perro persigue gato”), y un valor alto (p. ej., 0,9) representa una alta similitud semántica entre la primera secuencia 25SQ y la segunda secuencia 25SQ (p. ej., “Me gusta la torta”, “Disfruto las galletas”).

De acuerdo con algunas modalidades, el módulo de similitud 150 puede calcular un segundo valor de similitud vectorial 150S (p. ej., una inversa de la distancia coseno) que puede representar similitud entre

HOJA DE SUSTITUCIÓN (REGLA 26)

los vectores de incrustación de secuencias provisionales 140SE de la primera secuencia 25SQ y la segunda secuencia 25SQ.

El módulo de entrenamiento 120 puede calcular un
5 segundo valor de función de pérdida 120LF (denotado 120LF2), que representa una diferencia entre el segundo valor de similitud vectorial 150S y la similitud semántica de secuencias, como se representa mediante el primer elemento de datos de anotación 29AN. Por ejemplo,
10 el valor de función de pérdida 120LF (120LF2) puede estar en el rango de $[0, 1]$, donde un valor numérico grande (p. ej., 0,9) representa una gran diferencia entre el valor de similitud vectorial 150S y el elemento de datos de anotación 29AN, y donde un valor numérico pequeño (p.
15 ej., 0,1) representa una pequeña diferencia entre el valor de similitud vectorial 150S y el elemento de datos de anotación 29AN.

El módulo de entrenamiento 120 puede entonces entrenar el modelo de ML 110 para minimizar el segundo
20 valor de la función de pérdida 120LF (120LF2).

En otras palabras, el módulo de entrenamiento 120 puede utilizar el elemento de datos de anotación 29AN como información de supervisión para entrenar el modelo de ML 110. Por ejemplo, el módulo de entrenamiento 120
25 puede entrenar el modelo ML 110 al aplicar un algoritmo de retropropagación como se conoce en la técnica, para

ajustar pesos de NN 110 disminuyendo de esta manera la diferencia entre el valor de similitud 150S y el elemento de datos de anotación 29AN, y minimizando el valor de función de pérdida 120LF (120LF2).

5 Adicional, o alternativamente, el módulo de entrenamiento 120 puede entrenar el modelo ML 110 basado en pasajes anotados 25PT. En tales modalidades, los datos de anotación 29AN pueden ser, o pueden incluir una indicación (p. ej., 0/1), que define la existencia o
10 ausencia de una frase 27PH (ya sea como una copia exacta o como una versión semánticamente equivalente de esta) en un pasaje 25PT de interés.

Puede apreciarse que al representar el texto 25PT como incrustaciones 140SE de secuencias anidadas
15 jerárquicas 120HS, el sistema 10 puede simplificar el entrenamiento, y mejorar la inferencia del modelo 110: A diferencia de los métodos actualmente disponibles para extraer frases y temas de interés en un texto dado, el sistema 10 no requerirá la definición de una ubicación o
20 un marco donde se deba buscar una variación de la frase de interés. En ese sentido, la producción de secuencias anidadas jerárquicas 120HS puede verse intuitivamente como que facilita una búsqueda de un marco relevante dentro del texto 25PT, que abarcará el significado
25 semántico del tema, o frase de interés.

Por lo tanto, el elemento de datos de anotación de pasaje 29AN puede estar desprovisto de cualquier información que indique una ubicación de la variación de la frase de interés 27PH dentro del pasaje 25PT.

5 Como se elabora en la presente, el sistema 10 puede obtener del pasaje 25PT al menos un conjunto jerárquico 120HS de secuencias. De acuerdo con algunas modalidades, el módulo de capacitación 120 puede inferir el modelo basado en ML 110 en el conjunto jerárquico 10 120HS de secuencias (p. ej., una secuencia a la vez), para calcular, o encontrar un valor máximo de similitud vectorial provisional 150S. Como se explica en la presente, el valor máximo de similitud vectorial provisional 150S puede corresponder a una secuencia 25SQ 15 (en el conjunto jerárquico 120HS) que es más semánticamente similar a la frase 27PH, en ese punto en el proceso de entrenamiento.

El módulo de entrenamiento 120 puede entonces entrenar el modelo de ML 110 de tal manera que el valor 20 máximo de similitud vectorial provisional corresponda a, o represente la ocurrencia de la frase de interés 27PH o una variación semántica de esta en el pasaje 25PT, como se representa mediante la anotación de pasaje 29AN.

En otras palabras, el módulo de entrenamiento 25 120 puede utilizar el elemento de datos de anotación 29AN como información de supervisión para entrenar el modelo

de ML 110. Por ejemplo, el módulo de entrenamiento 120 puede entrenar el modelo ML 110 al aplicar un algoritmo de retropropagación como se conoce en la técnica, para ajustar pesos de NN 110, y de esta manera ajustar el
5 valor de similitud 150S para predecir la existencia o inexistencia de la frase 27PH (o una variante semántica de esta) en el texto de pasaje 25PT, como se indica mediante el elemento de datos de anotación 29AN.

Ahora se hace referencia a la Figura 5, que es
10 un diagrama de flujo que representa un método para identificar una variación de una frase en un pasaje de texto mediante al menos un procesador (p. ej., procesador 2 de la Figura 1), de acuerdo con algunas modalidades de la invención.

15 Como se muestra en la etapa S1005, el al menos un procesador 2 puede recibir (p. ej., a través de la entrada 7 de la Figura 1) u obtener una representación textual (p. ej., 27PH de la Figura 2) de la frase de interés. Basado en 27PH, el al menos un procesador 2
20 puede calcular un vector de incrustación de frases (p. ej., 140PE de la Figura 2), que representa un significado semántico de la frase 27PH, como se explica en la presente.

Como se muestra en la etapa S1010, el al menos
25 un procesador 2 puede obtener, o recibir (p. ej., a través de la entrada 7 de la Figura 1) una representación

textual del pasaje (p. ej., 25PT de la Figura 2), que puede incluir una pluralidad de palabras 25W, tokens o n-gramas.

Como se muestra en la etapa S1015, el al menos
5 un procesador 2 puede extraer, de la representación textual del pasaje 25PT, al menos un conjunto jerárquico (p. ej., 120HS de las Figuras 2, 3) de secuencias anidadas (p. ej., 25SQ de las Figuras 2, 4) de palabras 25W (o tokens o n-gramas), donde una o más (p. ej., cada)
10 secuencia de un conjunto jerárquico 120HS pueden formar un subconjunto de palabras 25W de una secuencia posterior 25SQ en el conjunto jerárquico.

Como se muestra en la etapa S1020, para cada secuencia 25SQ, el al menos un procesador 2 puede
15 calcular un vector de incrustación de secuencias correspondiente (p. ej., 140SE de las Figuras 2, 4), que representa un significado semántico de la secuencia respectiva 25SQ.

Como se muestra en la etapa S1025, para uno o
20 más vectores de incrustación de secuencias, el al menos un procesador 2 puede calcular un valor de similitud vectorial correspondiente (p. ej., 150S de la Figura 2), que representa la similitud de los vectores de incrustación de secuencias respectivos 140SE con el
25 vector de incrustación de frases 140PE.

Como se muestra en las etapas S1030 y S1035, el al menos un procesador 2 puede identificar una secuencia 25SQ correspondiente a un valor máximo de similitud vectorial 150S del uno o más valores de similitud vectorial, y posteriormente determinar la secuencia identificada 25SQ como una variación semántica de la frase 27PH, basado en el valor máximo de similitud vectorial 150S, p. ej., cuando el valor máximo de similitud vectorial 150S supera un umbral predeterminado.

10 Como se elabora en la presente, las modalidades de la invención pueden proporcionar una aplicación práctica para determinar una variación de una frase o tema en un pasaje, tal como una transcripción de una conversación hablada (p. ej., en un centro de llamadas).

15 La presente invención puede proporcionar una mejora en el campo tecnológico del procesamiento automatizado del lenguaje: Como se explica anteriormente, debido a una estructuración novedosa de datos semánticos (p. ej., incrustaciones de conjuntos jerárquicos 120HS),
20 las modalidades de la invención pueden encontrar de manera iterativa y segura porciones, o segmentos del pasaje que mejor correspondan semánticamente al tema o frase de interés. Este hallazgo no puede confinarse a ninguna restricción, definición o marco en el pasaje
25 examinado, como lo requieren los métodos actualmente disponibles de procesamiento del lenguaje natural.

A menos que se indique explícitamente, las modalidades del método descritas en la presente descripción no se limitan a un orden o secuencia particular. Además, todas las fórmulas descritas en la presente descripción están previstas solamente como ejemplos y pueden usarse otras fórmulas o fórmulas diferentes. Además, algunas de las modalidades del método o elementos de estas descritos pueden ocurrir o realizarse en el mismo punto en el tiempo.

Si bien se han ilustrado y descrito en la presente descripción ciertas características de la invención, muchas modificaciones, sustituciones, cambios y equivalentes pueden ocurrir a los expertos en la técnica. Por lo tanto, debe entenderse que las reivindicaciones adjuntas están destinadas a cubrir todas esas modificaciones y cambios que están dentro del verdadero espíritu de la invención.

Se han presentado varias modalidades. Cada una de estas modalidades puede incluir, por supuesto, características de otras modalidades presentadas, y las modalidades no descritas específicamente pueden incluir diversas características descritas en la presente descripción.