Dra. Elia Cano de Rovetto

CONTENIDO

Modulo III. Áreas de investigación en Inteligencia Artificial
Unidad V. Lenguaje Natural

Ilustración 1. Lenguaje Natural. live.staticflickr.com. CCO

El procesamiento del lenguaje natural (PNL) es un área de la inteligencia artificial que ayuda a las computadoras a comprender, interpretar y manipular el lenguaje humano. Este tiene gran aplicación en la actualidad. En el desarrollo de esta unidad se explicará cómo se realiza este proceso. Primero se explicará en qué consiste el procesamiento de señales. A continuación, se presenta cómo se realiza de la sintaxis y el análisis gramatical en PNL. La semántica y su significado es desarrollada en la sección 3. En la sección 4 se explica el concepto de pragmática y cómo se realiza. Finalmente, en la sección se desarrolla el tema de Generación de Lenguaje Natural.

El procesamiento del lenguaje natural (PNL) es un método de la Inteligencia Artificial para comunicarse con sistemas inteligentes que utilizan un lenguaje natural. Este implica hacer que las computadoras realicen tareas útiles con los lenguajes naturales que usan los humanos. La entrada y salida de un sistema PNL puede ser el habla o el texto escrito.

La investigación sobre PNL comenzó a principios de la década de 1950 después de la investigación de Booth & Richens y el memorándum de Weaver sobre traducción automática en 1949.

1954 fue el año en que se demostró un experimento limitado sobre traducción automática del ruso al inglés en el experimento Georgetown-IBM. En el mismo año, comenzó la publicación de la revista MT (Machine Translation). La primera conferencia internacional sobre traducción automática (MT) se celebró en 1952 y la segunda en 1956.

En 1961, el trabajo presentado en la Conferencia Internacional de Teddington sobre traducción automática de idiomas y análisis de idiomas aplicados fue el punto culminante de esta fase. A principios de 1961, el trabajo comenzó sobre los problemas de abordar y construir datos o bases de conocimiento. Este trabajo fue influenciado por la IA. En el mismo año, también se desarrolló un sistema de preguntas y respuestas de BÉISBOL. La entrada a este sistema estaba restringida y el procesamiento del lenguaje involucrado era simple.

Un sistema muy avanzado fue descrito en Minsky (1968). Este sistema, en comparación con el sistema de preguntas y respuestas de BÉISBOL, fue reconocido y proporcionó la necesidad de inferencia en la base de conocimiento para interpretar y responder a la entrada del lenguaje.

El enfoque gramatical-lógico, hacia el final de la década, ayudó con poderosos procesadores de oraciones de propósito general como el Motor del lenguaje central de SRI y la Teoría de la representación del discurso, que ofreció un medio para abordar un discurso más extenso.

El trabajo sobre el léxico en la década de 1980 también apuntó en la dirección del enfoque gramatical lógico. Podemos describir esto como una fase léxica y corpus. La fase tuvo un enfoque lexicalizado de la gramática que apareció a fines de la década de 1980 y se convirtió en una influencia creciente. Hubo una revolución en el procesamiento del lenguaje natural en esta década con la introducción de algoritmos de aprendizaje automático para el procesamiento del lenguaje.

5.1 Procesamiento de señales

El lenguaje natural pretende conseguir que una máquina comprenda lo que expresa una persona mediante el uso de una lengua natural. Las lenguas naturales o idiomas están basados usualmente en un sistema de signos a los cuales les denominaremos símbolos categóricos.

Los símbolos categóricos de un idioma pueden codificarse como una señal de comunicación de varias maneras: sonido, gesto, escritura, imágenes, etc. En el lenguaje humano se puede usar cualquiera de estos.

El procesamiento de señales toma las palabras habladas como entrada y las convierte en texto. En este proceso se debe primero limpiar los datos de texto no estructurados mediante la tokenización de palabras.

La tokenización es una de las tareas más comunes cuando se trata de trabajar con datos de texto. Esta consiste esencialmente en dividir una frase, oración, párrafo o un documento de texto completo en unidades más pequeñas, como palabras o términos individuales. Cada una de estas unidades más pequeñas se llaman tokens.

Los tokens pueden ser palabras, números o signos de puntuación. En la tokenización, las unidades más pequeñas se crean al ubicar los límites de las palabras. Estos son el punto final de una palabra y el comienzo de la siguiente palabra. Estos tokens se consideran como un primer paso para la derivación y la lematización.

Antes de procesar un lenguaje natural, necesitamos identificar las palabras que constituyen una cadena de caracteres. Es por eso que la tokenización es el paso más básico para proceder con datos de texto en PNL. Esto es importante porque el significado del texto podría interpretarse fácilmente analizando las palabras presentes en el texto.

5.2 Sintaxis y análisis gramatical

Las palabras son comúnmente aceptadas como las unidades más pequeñas de sintaxis. La sintaxis se refiere a los principios y reglas que rigen la estructura de las oraciones de cualquier idioma individual. Se centra en el orden adecuado de las palabras que pueden afectar su significado. Esto implica el análisis de las palabras en una oración siguiendo la estructura gramatical de la oración. Las palabras se transforman en la estructura para mostrar cómo se relacionan las palabras entre sí.

Procesar una oración sintácticamente implica determinar el sujeto y el predicado y el lugar de los sustantivos, verbos, pronombres, etc. Dado un léxico que le dice a la computadora la parte del discurso de una palabra, la computadora podría leer la frase de entrada palabra por palabra y al final producir una descripción estructural.

Sin embargo, podemos encontrar algunos problemas debido a que una palabra puede funcionar como diferentes partes del discurso en diferentes contextos (a veces un sustantivo, a veces un verbo, por ejemplo). También, puede haber varias interpretaciones posibles de la estructura de una oración.

Para implementar la tarea de análisis se utiliza el analizador, el cual es el componente de software diseñado para tomar datos de entrada (texto) y dar una representación estructural de la entrada después de verificar la sintaxis correcta según la gramática formal. También se construye una estructura de datos generalmente en forma de árbol de análisis o árbol de sintaxis abstracta u otra estructura jerárquica.

En las aplicaciones de análisis del lenguaje natural se utilizan las gramáticas de estructura de frase debido a que se caracterizan por aportar una especificación formal de una lengua, lo cual las hace aptas para ser implementadas mediante algoritmos computacionales. El analizador lo constituye el conjunto de reglas implementadas con el algoritmo que las ejecuta. Para obtener la cadena de entrada, necesitamos una secuencia de reglas de producción. A este conjunto de reglas de producción se le conoce como derivación.

Tipos de Análisis

1. Análisis de arriba hacia abajo: el analizador comienza a construir el árbol de análisis a partir del símbolo de inicio y luego intenta transformar el símbolo de inicio en la entrada. La forma más común de este análisis utiliza un procedimiento recursivo para procesar la entrada. La principal desventaja del análisis de descenso recursivo es el retroceso.

2. Análisis de abajo hacia arriba: el analizador comienza con el símbolo de entrada e intenta construir el árbol del analizador hasta el símbolo de inicio.

Durante el análisis, debemos decidir el no terminal, que se reemplazará junto con la decisión de la regla de producción con la ayuda de la cual se reemplazará el no terminal. Para decidir qué no terminal se reemplazará con la regla de producción se pueden usar **dos tipos de derivaciones**:

- **Derivación más a la izquierda:** en donde, la forma oracional de una entrada se escanea y se reemplaza de izquierda a derecha. La forma sentencial en este caso se llama forma sentencial izquierda.
- **Derivación más a la derecha:** en donde, se escanea la forma oracional de una entrada y se reemplaza de derecha a izquierda. La forma sentencial en este caso se llama forma sentenciosa derecha.

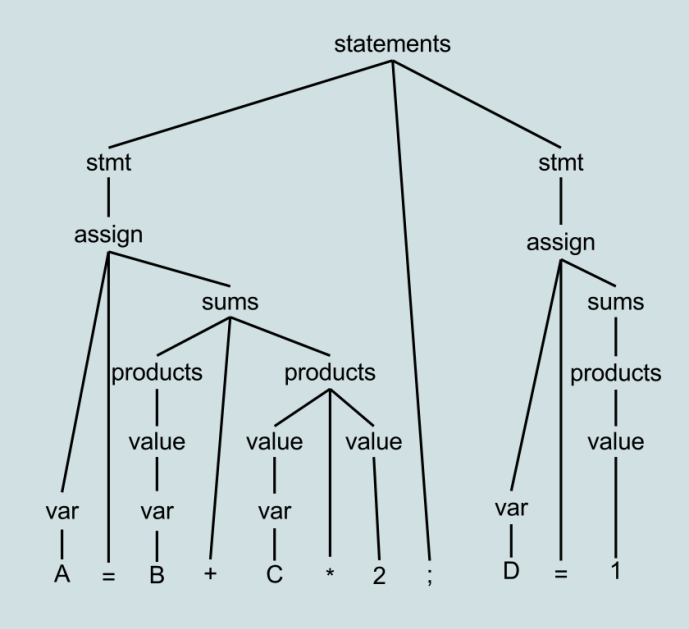


Ilustración 2. Ejemplo de árbol de análisis sintáctico. wikimedia.org. CCO

La derivación se puede representar gráficamente mediante el árbol de análisis. El símbolo de inicio de derivación sirve como la raíz del árbol de análisis. En cada árbol de análisis, los nodos hoja son terminales y los nodos interiores son no terminales. Una propiedad del árbol de análisis es que el recorrido en orden producirá la cadena de entrada original.

5.3 Semántica y significado

El análisis semántico es una estructura creada por el analizador sintáctico que asigna significados. Este componente transfiere secuencias lineales de palabras en estructuras y muestra cómo se asocian las palabras entre sí.

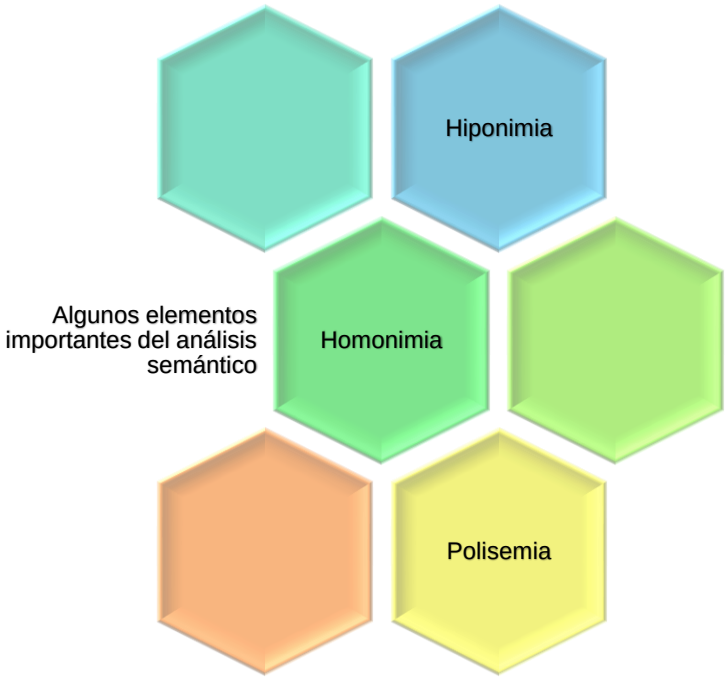
La semántica se enfoca solo en el significado literal de palabras, frases y oraciones. Esto solo abstrae el significado del diccionario o el significado real del contexto dado. Las estructuras asignadas por el analizador sintáctico siempre tienen un significado asignado.

El análisis semántico trata el significado de palabras y oraciones, las formas en que las palabras y oraciones se refieren a elementos en el mundo. El "significado" en estas discusiones generalmente se asocia con la semántica, pero en otros contextos he visto la sintaxis asociada con el "significado sintáctico".

El análisis semántico extrae el significado exacto o el significado del diccionario del texto y verifica el significado del texto. Se realiza mediante el mapeo de estructuras sintácticas y objetos en el dominio de la tarea. Esta primera parte del análisis semántico que estudia el significado de palabras individuales se llama semántica léxica. Incluye palabras, subpalabras, afijos (subunidades), palabras compuestas y frases. Se denominan colectivamente elementos léxicos a todas las palabras, subpalabras, etc. En otras palabras, podemos decir que la semántica léxica es la relación entre los elementos léxicos, el significado de las oraciones y la sintaxis de la oración.

Los siguientes son los pasos involucrados en la semántica léxica:

- La clasificación de elementos léxicos como palabras, subpalabras, afijos, etc. se realiza en semántica léxica.
- La descomposición de elementos léxicos como palabras, subpalabras, afijos, etc. se realiza en semántica léxica.
- También se analizan las diferencias y similitudes entre varias estructuras semánticas léxicas.



Hiponimia

es la relación entre un término genérico y las instancias de ese término genérico. Aquí el término genérico se llama hiperónimo y sus instancias se llaman hipónimos. Por ejemplo, la palabra color es hiperónimo y el color azul, amarillo, etc. son hipónimos.

Homonimia

Son las palabras que tienen la misma ortografía o la misma forma pero que tienen un significado diferente y no relacionado. Por ejemplo, la palabra "Murciélago" es una palabra de homonimia porque el murciélago puede ser un implemento para golpear una pelota o el murciélago también es un mamífero volador nocturno.

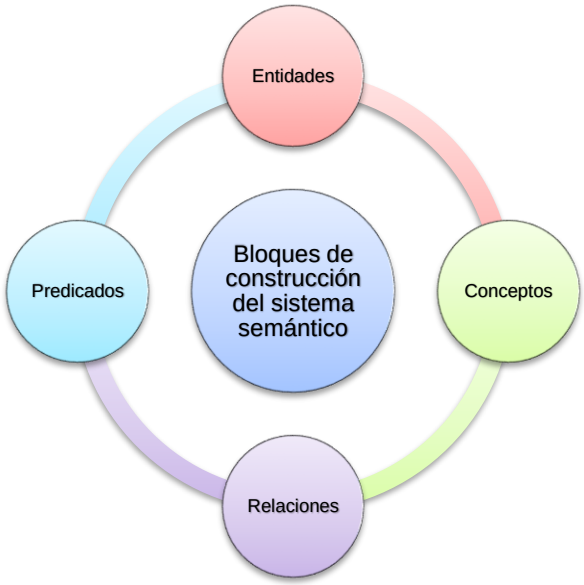
Polisemia

es una palabra griega, que significa "muchos signos". Es una palabra o frase con sentido diferente pero relacionado. En otras palabras, podemos decir que la polisemia tiene la misma ortografía, pero un significado diferente y relacionado. Por

ejemplo, la palabra "banco" es una palabra de polisemia que tiene los siguientes significados:

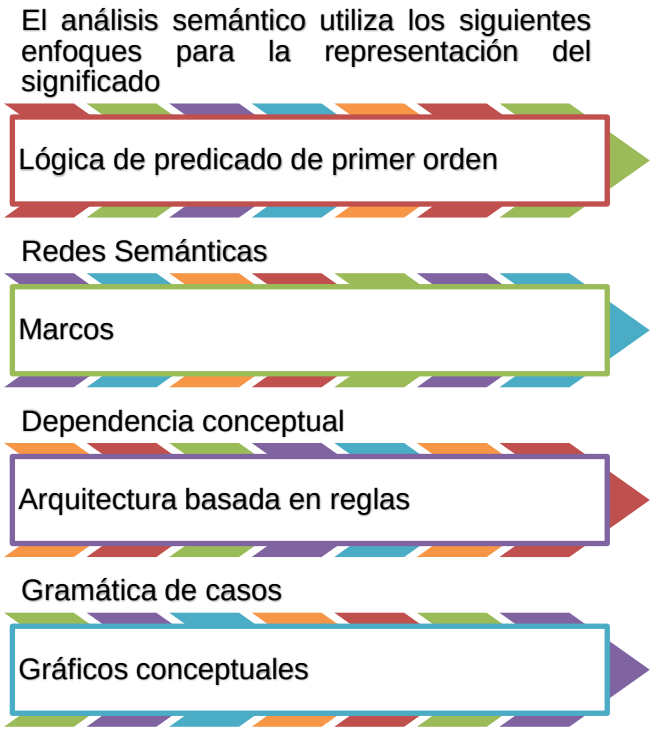
- Una institución financiera.
- Asiento que puede ser usado por varias personas a la vez.
- Conjunto de peces
- Conjunto de peces

El análisis semántico crea una representación del significado de una oración para eso utiliza los siguientes bloques de construcción del sistema semántico:



- **Entidades:** representa al individuo, como una persona en particular, ubicación, etc. Por ejemplo, Haryana. India, Ram, todas son entidades.
- **Conceptos:** representa la categoría general de los individuos, como una persona, ciudad, etc.
- **Relaciones:** Representa la relación entre entidades y concepto. Por ejemplo, Raúl es una persona.
- **Predicados:** representa las estructuras verbales. Por ejemplo, los roles semánticos y la gramática de casos son ejemplos de predicados.

Ahora, podemos entender que la representación del significado muestra cómo armar los bloques de construcción de los sistemas semánticos. En otras palabras, muestra cómo agrupar entidades, conceptos, relaciones y predicados para describir una situación. También permite el razonamiento sobre el mundo semántico.



5.4 Pragmática

El análisis pragmático es parte del proceso de extracción de información del texto. Específicamente, es la parte que se enfoca en tomar un conjunto de estructuras de texto y descubrir cuál era el significado real. Incorpora, así mismo, información sobre las relaciones que se dan entre los hechos que forman el contexto y entre diferentes entidades.

Además, añade información adicional al análisis del significado de la frase en función del contexto donde aparece. Se trata de uno de los niveles de análisis más

complejos, la finalidad es incorporar al análisis semántico la aportación significativa que pueden hacer los participantes, la evolución del discurso o información presupuesta.

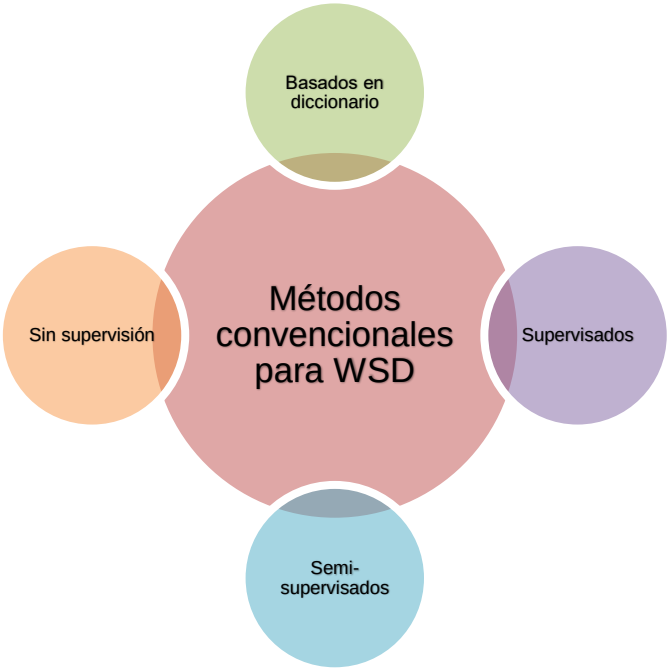
La ambigüedad léxica, sintáctica o semántica, es uno de los primeros problemas que enfrenta cualquier sistema de PNL. Los etiquetadores de parte del discurso (Part-of-speech – POS, en inglés) con un alto nivel de precisión pueden resolver la ambigüedad sintáctica de la palabra. Por otro lado, el problema de resolver la ambigüedad semántica se llama desambiguación del sentido de las palabras (Word Sense Disambiguation – WSD en inglés). Resolver la ambigüedad semántica es más difícil que resolver la ambigüedad sintáctica.

La capacidad de determinar qué significado de la palabra se activa mediante el uso de la palabra en un contexto particular se le denomina desambiguación. Asociar las palabras en contexto con su entrada más adecuada en un inventario de sentidos predefinido es la tarea de la Desambiguación del sentido de las palabras. El inventario de sentido de facto para inglés en WSD es WordNet. Por ejemplo, si usamos la palabra "mouse" en la siguiente oración: *"Un mouse consiste en un objeto sostenido en una mano, con uno o más botones"* asignaríamos "mouse" con su sentido de dispositivo electrónico (el 4to sentido en el inventario de sentidos de WordNet).

La evaluación de WSD requiere las siguientes dos entradas:

- **Un diccionario:** la primera entrada para la evaluación de WSD es el diccionario, que se utiliza para especificar los sentidos a ser desambiguados.
- **Test Corpus:** otra entrada requerida por WSD es el corpus de prueba de alta anotación que tiene el objetivo o los sentidos correctos. Los corpus de prueba pueden ser de dos tipos:

- **Muestra léxica:** este tipo de corpus se utiliza en el sistema, donde se requiere desambiguar una pequeña muestra de palabras.
- **Todas las palabras**: este tipo de corpus se usa en el sistema, donde se espera que desambigüe todas las palabras en un texto en ejecución.



Métodos basados en diccionario o basados en conocimiento

estos métodos se basan principalmente en diccionarios, tesoros y bases de conocimiento léxico. No utilizan evidencias corporales para la desambiguación. El método de Lesk es el método seminal basado en el diccionario introducido por Michael Lesk en 1986. La definición de Lesk, en la que se basa el algoritmo de Lesk, es "medir la superposición entre definiciones de sentido para todas las palabras en contexto". Sin embargo, en 2000, Kilgarriff y Rosensweig dieron la definición simplificada de Lesk como "superposición de medidas entre las definiciones de sentido de la palabra y el contexto actual", lo que significa además identificar el sentido correcto para una palabra a la vez. Aquí el contexto actual es el conjunto de palabras en la oración o párrafo circundante.

Métodos supervisados

Para la desambiguación, los métodos de aprendizaje automático utilizan cuerpos anotados con sentido para entrenar. Estos métodos suponen que el contexto puede proporcionar evidencia suficiente por sí solo para desambiguar el sentido. En estos métodos, las palabras conocimiento y razonamiento se consideran innecesarias. El contexto se representa como un conjunto de "características" de las palabras. Incluye la información sobre las palabras circundantes también. La máquina de vectores de soporte y el aprendizaje basado en la memoria son los enfoques de aprendizaje supervisado más exitosos para WSD. Estos métodos dependen de una cantidad sustancial de corpus etiquetados manualmente, lo cual es muy costoso de crear.

Métodos semi-supervisados

Debido a la falta de corpus de entrenamiento, la mayoría de los algoritmos de desambiguación de sentido de las palabras utilizan métodos de aprendizaje semi-supervisados. Esto se debe a que estos métodos utilizan tanto datos etiquetados como no etiquetados. Estos métodos requieren una cantidad muy pequeña de texto anotado y una gran cantidad de texto sin anotar. La técnica utilizada por los métodos semi-supervisados es el arranque a partir de datos de semillas.

Métodos sin supervisión

Estos métodos suponen que se producen sentidos similares en un contexto similar. Es por eso, que los sentidos pueden ser inducidos a partir del texto al agrupar las ocurrencias de palabras usando alguna medida de similitud del contexto. Esta tarea se llama inducción de sentido de palabra o discriminación. Los métodos no supervisados tienen un gran potencial para superar el cuello de botella de adquisición de conocimiento debido a la no dependencia de los esfuerzos manuales.

5.5 Generación de Lenguaje Natural

Es el proceso de producir frases y oraciones significativas en forma de lenguaje natural a partir de alguna representación interna.

Pasos de la Generación de Lenguaje Natural

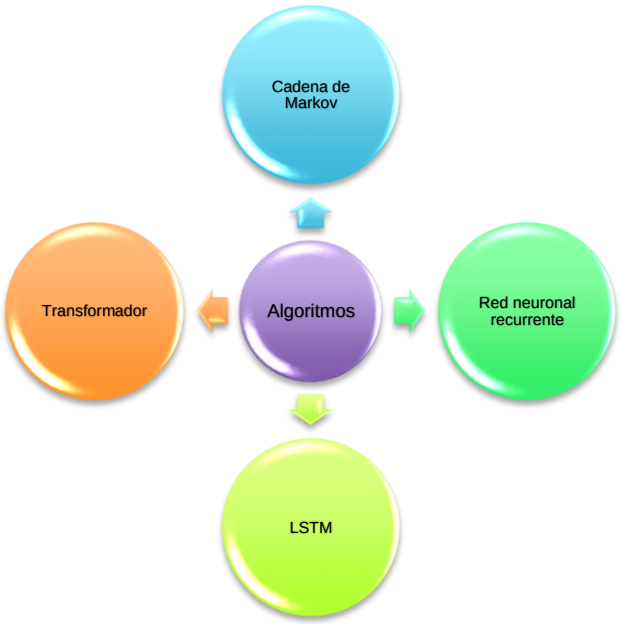
- **Planificación de texto:** incluye la recuperación del contenido relevante de la base de conocimiento.
- **Planificación de oraciones:** incluye elegir las palabras requeridas, formar frases significativas y establecer el tono de la oración.
- **Realización de texto:** está mapeando el plan de oración en la estructura de la oración

El generador de lenguaje natural ayuda a la máquina a clasificar a través de muchas variables para que los datos sean comprensibles y busca automatizar la escritura de narrativas basadas en dichos datos. Su objetivo es crear sistemas informáticos capaces de producir por sí mismos textos con sentido para el ser humano.

Son capaces de generar automáticamente narrativas que describen, resumen o explican los datos estructurados de entrada de manera humana a la velocidad de miles de páginas por segundo.

Existen dos enfoques principales para la generación de lenguaje: el uso de plantillas y la creación dinámica de documentos. Si bien solo se considera que este último es NLG "real", hubo una forma larga y de varias etapas de plantillas básicas y directas

a lo más avanzado y cada nuevo enfoque amplió la funcionalidad y las capacidades lingüísticas adicionales. La generación de lenguaje natural se basa en una serie de algoritmos que abordan ciertos problemas de creación de textos similares a los humanos.



Cadena de Markov: fue uno de los primeros algoritmos utilizados para la generación del lenguaje. Este modelo predice la siguiente palabra en la oración usando la palabra actual y considerando la relación entre cada palabra única para calcular la probabilidad de la siguiente palabra. De hecho, los ha visto mucho en versiones anteriores del teclado del teléfono inteligente, donde se utilizaron para generar sugerencias para la siguiente palabra en la oración.

Red neuronal recurrente (RNN): son modelos que intentan imitar el funcionamiento del cerebro humano. Los RNN pasan cada elemento de la secuencia a través de una red de alimentación directa y utilizan la salida del modelo como entrada para el siguiente elemento de la secuencia, lo que permite almacenar la información del paso anterior. En cada iteración, el modelo almacena las palabras

anteriores encontradas en su memoria y calcula la probabilidad de la siguiente palabra. Para cada palabra en el diccionario, el modelo asigna una probabilidad basada en la palabra anterior, selecciona la palabra con la probabilidad más alta y la almacena en la memoria. La "memoria" de RNN hace que este modelo sea ideal para la generación de lenguaje porque puede recordar el fondo de la conversación en cualquier momento. Sin embargo, a medida que aumenta la longitud de la secuencia, los RNN no pueden almacenar palabras que se encontraron de forma remota en la oración y hace predicciones basadas solo en la palabra más reciente. Debido a esta limitación, los RNN no pueden producir oraciones largas coherentes.

LSTM: para abordar el problema de las dependencias de largo alcance, se introdujo una variante de RNN llamada Long long-term memory (LSTM). Aunque similar a RNN, los modelos LSTM incluyen una red neuronal de cuatro capas. El LSTM consta de cuatro partes: la unidad, la puerta de entrada, la puerta de salida y la puerta olvidada. Estos permiten que el RNN recuerde u olvide palabras en cualquier intervalo de tiempo ajustando el flujo de información de la unidad. Cuando se encuentra un punto, la puerta olvidada reconoce que el contexto de la oración puede cambiar y puede ignorar la información del estado de la unidad actual. Esto permite que la red rastree selectivamente solo información relevante mientras minimiza el problema de gradiente que desaparece, lo que permite que el modelo recuerde información durante un período más largo.

Aún así, la capacidad de la memoria LSTM está limitada a unos cientos de palabras debido a sus rutas secuenciales inherentemente complejas desde la unidad anterior a la unidad actual. La misma complejidad resulta en requisitos computacionales altos que hacen que LSTM sea difícil de entrenar o paralelizar.

Transformador: un modelo relativamente nuevo se introdujo por primera vez en el documento de Google de 2017 "La atención es todo lo que necesita", que propuso

un nuevo método llamado "mecanismo de auto atención". El transformador consiste en una pila de codificadores para procesar entradas de cualquier longitud y otro conjunto de decodificadores para generar las oraciones generadas. A diferencia de LSTM, el Transformador realiza solo un número pequeño y constante de pasos, mientras aplica un mecanismo de auto atención que simula directamente la relación entre todas las palabras en una oración. A diferencia de los modelos anteriores, el Transformador utiliza la representación de todas las palabras en contexto sin tener que comprimir toda la información en una única representación de longitud fija que le permite al sistema manejar oraciones más largas sin que se disparen los requisitos computacionales.

Uno de los ejemplos más famosos del Transformador para la generación de lenguaje es OpenAI, su modelo de lenguaje GPT-2. El modelo aprende a predecir la siguiente palabra en una oración al enfocarse en las palabras que se vieron previamente en el modelo y que están relacionadas con la predicción de la siguiente palabra. Una actualización más reciente de Google, la representación del codificador bidireccional Transformers (BERT), proporciona los resultados más avanzados para diversas tareas de PNL.