1/4/23 - **Introducción a la desambiguación**

|  |  |
| --- | --- |
| Puntos clave | Notas |
| **Cadenas de Markov**  Sirven para calcular escenarios.  Escalera de modelos  Machine Learning clásico:   * Hidden Markov Models * Maximun Entropy Markov Models   Deep Learning:   * Averaged Perceptron Tagger | **Introducción a la desambiguación**  El lenguaje humano es difuso y ambiguo y requiere mucho contexto.  Ambigüedades del lenguaje   * Por agrupamiento * Funcional * Léxica   **Cadenas de Markov**   * contiene un conjunto fino de estados, debe tener un conjunto de etiquetas bien definidas. * Define probabilidades de transición entre los posibles estados que una categoría puede tener.   En una cadena de Márkov se necesitan los siguientes elementos:   1. matrices de transición 2. distribución inicial de estados   **Algoritmo:** “modelo markoviano latente”  Tokenizador Punkt: hace referencia a una palabra alemana relacionada con puntuación. Es el algoritmo de tokenizacion estándar.  Average Perceptron Tagger: es un etiquetador que comparte una noción muy similar al de las redes neuronales.  Tiene una eficiencia superior a los modelos etiquetado de máxima entropía.  Es el etiquetador por defecto en el idioma inglés, teniendo un algoritmo pre-entrenado para eso.  **Modelos Markovianos Latentes**   * Las cadenas de markov son la base para entender cómo funcionan los etiquetadores de palabras. * En una cadena de markov tenemos dos ingredientes principales: * Matriz de transición (cada elemento representa la probabilidad de transición al siguiente estado) |
| Sumario: | |

04/07/23 – Modelos Markovianos Latentes

|  |  |
| --- | --- |
| Puntos clave | Notas |
|  | Distribución de estados  HMM_1  Ejemplo: Consideramos una cadena de Markov donde tenemos 3 estados(1: frio, 2: caliente, 3:tibio).  HMM_2  Calculamos la probabilidad del evento 2 dado 3, primero calculamos la probabilidad P(3:tibio,2:caliente), esto es la probabilidad de que transicionemos de un estado 3 a un estado 2, (contamos los eventos y tenemos solo 1 evento posible, y lo dividimos entre la cantidad de eventos totales posibles (5 eventos)).  **P(3,2) = 1/5**  Ahora contamos la cantidad de dias tibios en nuestros datos, y lo dividimos entre la cantidad de dias. **P(3) = 1/5**  Por tanto la probabilidad P(2|3) = 1, y estará representada en la matriz con la celda C32. Ahora replicamos los pasos para construir la matriz. |
| Sumario: | |

04/08/23 -

|  |  |
| --- | --- |
| Puntos clave | Notas |
|  | Calculamos el resto de la matriz, para realizar mis cálculos los realice a mano aunque no digitalice la hoja, doy por sentado que en la imagen la última letra C está de más, esto porque hablamos de temperatura tomada en un espacio de 5 días, en las llaves de temperaturas existen 6 datos por lo cual para mí fue de la siguiente manera  {d1, d2, d3, d4, d5} -> {f, f ,t ,c , c}  5 dias -> 5 mediciones de temperatura   * ya con la matriz de estados consideramos una distribución de estados inicial. * Los estados de markov se definen solo por el estado inmediatamente anterior. * Como vimos en la clase inicial al multiplicar la matriz A por el estado inicial me dará la probabilidad de ocurrencia de los estados par el dia siguiente y obtenemos el siguiente vector PI(1).   Y esta es la fórmula general de la Cadena de Markov.  Verificamos haciendo la multiplicación elemento a elemento.  EN general cuanto tienes estados en un tiempo t, y lo multiplicas por la matriz de estados da como resultado la probabilidad de cada estado al día siguiente, y esta es la fórmula general de una cadena de markov.  Utilizando el producto cruz (o producto punto) hacemos la multiplicación de ambas matrices y obtenemos el resultado para la probabilidad de los estados para el dia siguiente (Vector PI(1)).  HMM_10 |
| Sumario: | |

1/4/23 **– Cadenas de Markov Latentes -** **El algoritmo de Viterbi**

|  |  |
| --- | --- |
| Puntos clave | Notas |
| **Algoritmo Viterbi**  El proceso mediante el cual un **Modelo Markoviano Latente** determina la secuencia de etiquetas más probable para una secuencia de palabras es:  el producto de a probabilidad inicial de encontrar esa etiqueta en la primera parte de esa secuencia este multiplicado por una probabilidad condicional. | El objetivo de una cadena de markov latente es encontrar dada una secuencia de palabras cuál es la secuencia de etiquetas que le corresponde con mayor probabilidad  Otra definición pudiese ser: El objetivo de una cadena de Markov Latente es encontrar cuál es la secuencia de etiquetas que le corresponde, con mayor probabilidad, a una secuencia de palabras dada  **Ingredientes de la cadena de markov latente:**   * Matriz de transición * Distribución de estados * Probabilidades de emisión   **El algoritmo de Viterbi**  La pregunta es **¿qué tipo de predicciones hace un modelo markoviano latente HMM)?**  viterbi_1  Una vez entrenado, el proceso que denominaremos Decodificación consiste en que dada una secuencia de palabras podamos identificar la secuencia de etiquetas gramaticales más probable que le corresponda, y esto se hace mediante el algoritmo de Viterbi.  Hay otras alternativas pero primero enfoquemos en este.  La parte de entrenamiento que programamos consiste en encontrar la matriz A con sus coeficientes C y luego las probabilidades Emisión que son los B dados las probabilidades condicionales (word|tag)... |
| Sumario: | |

04/08/23

|  |  |
| --- | --- |
| Puntos clave | Notas |
|  | luego viene el algoritmo de Viterbi que se va a encargar de encontrar de entre un montón de secuencias la secuencia más probable esto lo hace asignándole una probabilidad a cada secuencia que llamaremos probabilidad de Viterbi, luego dentro de ese espacio de probabilidades escogemos la mayor y esa sería la que el algoritmo va a retornar como la más probable y por lo tanto la que debería ser las etiquetas correctas de la secuencia de palabras.  **El algoritmo de Viterbi funciona de la siguiente manera**:  viterbi_2  Cada columna son todas las posibles etiquetas que una palabra va a tener, castillo es una persona no un edificio un sustantivo, cada circulo corresponde a una posible categoría gramatical, los círculos en gris tienen una probabilidad cero.  ¿**Como esto nos ayuda a entender el algoritmo de Viterbi**? de la siguiente manera.  viterbi_3 |
| Sumario: | |

04/08/23

|  |  |
| --- | --- |
| Puntos clave | Notas |
|  | Considerando todas las posibles categorías gramaticales de cada palabra vamos creando caminos creando las etiquetas posteriores, cada camino es recorrer la primera etiqueta posible hasta la siguiente posible y así hasta llegar a la última palabra que contiene la secuencia. De todos esos caminos hay que calcular el más probable, eso lo hacemos calculando un numero probabilístico que me diga que tan probable es que sea uno de esos caminos y escoger el mayor, eso lo hacemos de la siguiente manera.  viterbi_4  El circulo de sustantivo propio está en color verde, significa que vamos a analizar lo que sucede en este nodo en particular, vamos a denotar con la letra griega NU, que parece una letra V paréntesis prop estilizada probabilidad de viterbi de que la categoría gramatical de castillo sea "PROP" y eso es igual al "producto de la probabilidad inicial" este multiplicado por una "probabilidad condicional" de que ya que la categoría Inicial es PROP la palabra que este ubicada ahí es castillo, ese cálculo lo hacemos para cada una de las celdas de esta columna, de esta manera calculamos la probabilidad de Viterbi para cada uno de los nodos de la primera columna.  viterbi_5 viterbi_6 |
| Sumario: | |

1/4/23 - **Calculo de las probabilidades de Viterbi**

|  |  |
| --- | --- |
| Puntos clave | Notas |
|  | El único nodo que tiene probabilidad no nula en la segunda columna es el de categoría determinante (DET), para calcular la probabilidad de este nodo lo que vamos a hacer es considerar todos los posibles caminos que pasan por ese nodo, vemos que son dos NOUN-DET Y PROP-DET, cada uno de los números tiene una probabilidad asignada que consiste en tomar la probabilidad del estado anterior (V1(PROP)), multiplicar por la probabilidad condicional de la etiqueta anterior y cual será la etiqueta siguiente que en este caso será de que dado prop la siguiente sea DET y esto multiplicado por una probabilidad de emisión, que dado que la categoría es DET cual será la palabra el, y que tan probable es eso, aplicamos lo mismo a otra ruta con NOUN, y de esos dos números calculamos la probabilidad y tomamos esa como la nueva probabilidad de Viterbi del nodo que está en verde en este momento, y así subsecuentemente para todos los nodos en la columna, el proceso estará completo cuando hayamos calculado las probabilidades de cada uno de los elementos de esta matriz.  **Cálculo de las probabilidades de Viterbi**   * el **algoritmo de Viterbi** asigna probabilidades a cada elemento en una matriz que combina filas de categorías gramaticales y columnas de palabras en una secuencia. * la idea es que con este algoritmo y con estas probabilidades puede determinar cuál es la secuencia de etiquetas más probable para esa secuencia que nosotros le estamos dando al modelo como entrada. * El **algoritmo de viterbi** termina cuando hemos calculado las probabilidades de todos los elementos de una matriz.   En resumen el algoritmo de Viterbi mediante la búsqueda de posibles caminos de etiquetas calcula:   * una probabilidad a cada elemento de una matriz donde esas probabilidades las llamamos probabilidades de Viterbi. * el objetivo de encontrar la secuencia más probable consiste en encontrar el camino cuyas probabilidades de Viterbi son más grandes. |
| Sumario: | |

1/4/23 - **Modelos Markovianos de máxima entropía (MEMM)**

|  |  |
| --- | --- |
| Puntos Clave | Notas |
|  | El Modelo de Markov de Máxima Entropía es un modelo gráfico discriminativo, basado en regresión logística multinomial, que permite el etiquetado de secuencias y es utilizado para el POS Tagging. … Fi(o,q) indica una función característica sobre una propiedad de la observación. |
| Sumario: | |

1/4/23

|  |  |
| --- | --- |
| Key Points | Notes |
|  |  |
| Summary: | |

1/4/23

|  |  |
| --- | --- |
| Key Points | Notes |
|  |  |
| Summary: | |