

Algorítmica

Curso 2023-2024

Grupo Viterbi



PRÁCTICA 5- PROGRAMACIÓN DINÁMICA

Integrantes:

Miguel Ángel De la Vega Rodríguez	miguevrod@correo.ugr.es
Alberto De la Vera Sánchez	joaquinrojo724@correo.ugr.es
Joaquín Avilés De la Fuente	adelaveras01@correo.ugr.es
Manuel Gomez Rubio	e.manuelgmez@go.ugr.es
Pablo Linari Perez	e.pablolinari@go.ugr.es

Facultad de Ciencias UGR
Escuela Técnica Ingeniería Informática UGR
Granada
2023-2024

Índice general

1	Autores	3
2	Equipo de trabajo	4
3	Ejemplos de Uso	5
3.1	Lenguaje	5
3.1.1	Desambiguación de palabras	5

Autores

- **Miguel Ángel De la Vega Rodríguez: 20%**
 - Estructura del documento
 - Ejemplos de Uso
- **Joaquín Avilés De la Fuente: 20%**
 - RELLENAR
- **Alberto De la Vera Sánchez: 20%**
 - RELLENAR
- **Manuel Gomez Rubio 20%**
 - RELLENAR
- **Pablo Linari Pérez: 20%**
 - RELLENAR

Apartado 2

Equipo de trabajo

- **Miguel Ángel De la Vega Rodríguez:** (Ordenador donde se ha realizado el compute)
 - AMD Ryzen 7 2700X 8-Core
 - 16 GB RAM DDR4 3200 MHz
 - NVIDIA GeForce GTX 1660 Ti
 - 1 TB SSD NVMe
 - Debian 12 Bookworm
 - Compilador GCC 12.2.0

Ejemplos de Uso

El algoritmo de Viterbi es ampliamente utilizado en diversos campos tales como el lenguaje, la ingeniería de comunicaciones, la robótica, la biología, la medicina, la meteorología, etc. A continuación presentamos algunas de las aplicaciones específicas de este algoritmo en distintos campos.

3.1 Lenguaje

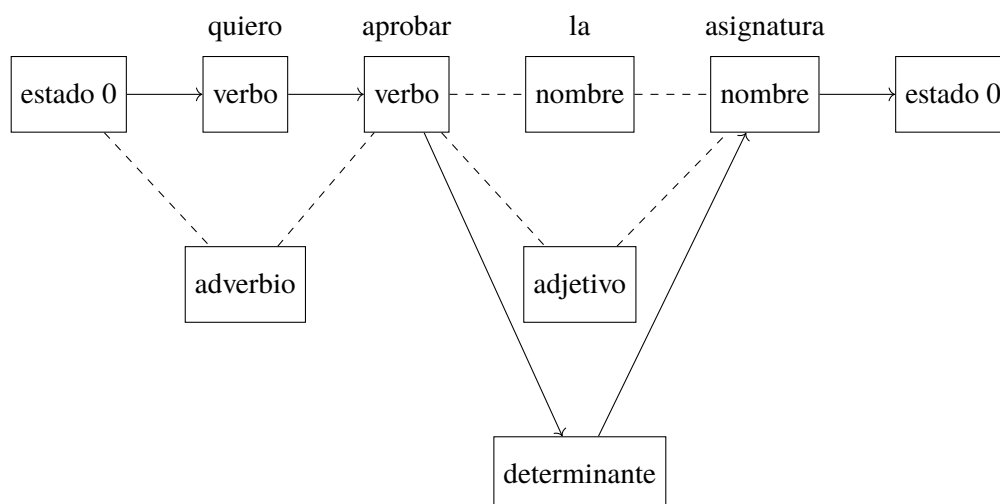
El algoritmo de Viterbi es utilizado en el reconocimiento de voz, en el reconocimiento de escritura a mano, en la corrección de errores en el texto, en la traducción automática, en la generación de texto, en la síntesis de voz, en la transcripción de audio, etc. Veamos un ejemplo de uso de Viterbi en el proceso de desambiguación de palabras en un texto.

3.1.1 Desambiguación de palabras

Cuando queremos procesar el lenguaje natural, es común encontrarnos con palabras que tienen múltiples significados. Por ejemplo, la palabra *privado* está reconocida por la RAE como un adjetivo, un sustantivo y un verbo. Para desambiguar estas palabras cuando procesamos un texto, podemos utilizar el algoritmo de Viterbi, en este caso particular, los elementos del modelo oculto de Markov serían:

- El conjunto Q de estados ocultos (categorías gramaticales)
- El conjunto V de estados observables (palabras)
- El conjunto A de probabilidades de transición entre estados (probabilidades de cambio de categoría gramatical, por ejemplo, que un nombre vaya detras de un verbo)
- El conjunto B de probabilidades de emisión de observaciones (probabilidades de que una palabra pertenezca a una categoría gramatical, por ejemplo, que la palabra *perro* sea un sustantivo es mucho mayor a que sea un adjetivo)

Mostramos un ejemplo para el texto *quiero aprobar la asignatura*:



Donde, los observables son la secuencia de palabras *quiero aprobar la asignatura* y los estados ocultos son las categorías gramaticales de las palabras, que como se puede ver, contemplan sólo un conjunto limitado de categorías gramaticales. Esto se debe a que la probabilidad de pertenencia de determinadas palabras a ciertas categorías gramaticales es 0, lo que simplifica y acelera el proceso de desambiguación.

Aplicación del Algoritmo de Viterbi

El algoritmo de Viterbi se utiliza para encontrar la secuencia más probable de estados ocultos (categorías gramaticales) dada una secuencia de observaciones (palabras). Para nuestro ejemplo, el proceso se desarrolla de la siguiente manera:

- Inicialización:
 - Definir los estados ocultos posibles. Por ejemplo, en nuestro caso, los estados posibles son verbo, nombre, determinante, etc.
 - Definir las probabilidades iniciales para cada estado oculto. Por ejemplo, es muy probable que la primera palabra *quiero* sea un verbo.
 - $\pi(\text{verb}) = P(\text{verb}|\text{estado}_0)$.
- Recursión:
 - Para cada palabra en la secuencia, calcular la probabilidad de que cada posible estado oculto (categoría gramatical) siga a cada estado anterior. Esto se hace utilizando las probabilidades de transición (A) y las probabilidades de emisión (B).
 - Ejemplo: Para la palabra *aprobar*, se calcularía $P(\text{verb}_2|\text{verb}_1) \cdot P(\text{aprobar}|\text{verb}_2)$ para todas las categorías posibles de aprobar.
 - Se selecciona el estado que maximiza esta probabilidad.
- Terminación:
 - Una vez procesadas todas las palabras, se selecciona la secuencia de estados que maximiza la probabilidad total de la secuencia observada.
 - Esta secuencia representará las categorías gramaticales más probables para la oración.
- Reconstrucción de la secuencia de estados:

-
- Utilizando las probabilidades calculadas, se reconstruye la secuencia de categorías gramaticales más probable.
 - Por ejemplo: quiero (verbo), aprobar (verbo), la (determinante), asignatura (nombre).