Procesamiento de lenguaje natural y análisis de textos

Gonzalo Rivero

Big Data en investigación social y opinión pública

22 a 26 de julio, 2019

Introducción

- Tenemos acceso a enormes cantidades de data en forma de texto
  - Artículos de prensa
  - Textos legales
  - Correos electrónicos
  - Informes y documentos oficiales
- En esta sesión intentaremos analizar ese tipo de información cuantitativamente
  1. Extraer información de los textos
  2. Análizar cuantitativamente los textos

Tareas del análisis de texto

- Resumir bloques de texto para extraer la información relevante e ignorar datos superfluos
- Crear bots conversacionales
- Generar etiquetas para descubrir temas en cuerpos de texto
- Identificar tipo de entidades en el texto como personas, lugares, organizaciones o fechas
- Clasificar textos de acuerdo con su "sentimiento" o polaridad afectiva para saber si hablan positiva o negativamente de algo

Procesamiento del lenguaje natural

- Extraer información a partir del texto, *semánticamente* si es posible
- Dado un texto responder con de quién habla, qué hace esa persona, ...
- Diferentes niveles de complejidad:
  1. Separar un texto en unidades (tokenizar)
  2. Transfomar un término a su versión de diccionario
  3. Identificar personas, organizaciones, fechas, ... en un texto
  4. Reconocer partes gramaticales de una frase

Tokenización

- Tokenizar es el proceso de romper cadenas de texto en palabras, frases u otros elementos con sentido que llamaremos *tokens*. Pero
  - Star Wars: Dos palabras, no en diccionario, pero un solo concepto
- Podemos usar expresiones regulares para dividir en piezas que no son alfanuméricas, como espacios, puntuación, hashtags, ...
- Fallará en caso de términos con varias palabras que se refieren a una unidad léxica como Nueva York o algunas expresiones.
- n-grams (permitir trabajar con 'Nueva' 'York' y 'Nueva York')
- Podemos usar modelos de Machine Learning que estimen si $\Pr(word_i|word_{i-1})$ es lo bastante alta para considerarlo una unidad

Normalización

- Proceso de normalización de los tokens. Reducir todas las instancias a una clase de equivalencia léxica
  1. plurales $\rightarrow$ singular (gatos $\rightarrow$ gato)
  2. géneros $\rightarrow$ "neutro" (gata $\rightarrow$ gato)
  3. pasados $\rightarrow$ presente (cantaba $\rightarrow$ cantar)
  4. adverbios $\rightarrow$ adjetivo (llanamente $\rightarrow$ llano)

Tipos de normalizaciones

1. Reducir a versión de dicionario mediate lematización. Requiere un parseador morfológico que reconozca en un término cosas como sujeto, tiempo verbal, modificadores, ...
2. Parsers usan dicionario de raices y afijos con metadatos (como POS) así como algunas reglas (las -s de plural van después de nombres).
3. *Stemmer* solo elimina datos morfológicos (la 's' del plural, las terminaciones de los verbos regulares). Muchos errores y no crea palabras existentes pero es suficiente ('educación' a 'educ').

Reconocimiento de entidades

- Identificar y clasificar nombres en el texto

  U.S. Rep. Ernest Istook, R-Warr Acres, is scheduled to conduct a town hall meeting about Internet safety for children at 6:30 p.m. Thursday at Putnam City High School, 5300 NW 50.

  Entidad política Persona Organización Hora Fecha

Aplicaciones del reconocimiento de entidades

- Indexar y fusionar entidades
- Asociar sentimiento a nombres de compañías o productos
- Muchas tareas de extracción de información dependen de asociación entre entidades
- Responder preguntas en un chat/bot requiere reconocer entidades

Identificar entidades

- Puede hacerse con expresiones regulares y aprovechar POS (saber si una palabra es verbo, sustantivo, ...), identifación de frases, categorías semánticas (que reduzcan sinónimos a un término), ...
- Literatura moderna favorece modelos en secuencia
  1. Recoger documentos representativos
  2. Etiquetar cada token de acuerdo con su entidad
  3. Diseñar variables que capturen el texto y la secuencia de clases
  4. Entrenar un HMM o RNN que prediga las etiquetas
- Usan diccionarioes sobre la palabra actual, la siguiente, POS y el contexto de la etiqueta
- Tarea relacionada es coreferencia
  - John F Kennedy es Kennedy

Etiquetadores de partes del discurso

- Marcar cada pieza del discurso como nombre, verbo, preposición, ...
- Analíticamente es igual a reconocimiento de entidades
- Modelo puede incluir
  - La palabra actual, la siguiente, la anterior, ...
  - La etiqueta de esas palabras
  - Información sobre la palabra misma como sufijos

Análisis de textos

Análisis de temas

Gonzalo Rivero

Big Data en investigación social y opinión pública

22 a 26 de julio, 2019

- Queremos interpretar el contenido de una colección grande de documentos
  - ¿Qué temas de investigación son más exitosos al pedir financiación?
  - ¿Qué temas llevan a discusiones entre los editores de Wikipedia?
  - ¿Qué temas preocupan a cada político? ¿Cómo cambian los temas de interés a lo largo del tiempo?
- Analizaremos textos de forma no supervisada para poder
  - Agrupar documentos según el tema del que hablen
  - Asociaremos términos a cada uno de esos temas

Cuantificar documentos

- *Bag of words* representa el documento como una tabla de frecuencia de cada palabra.
- El documento "Prefiero las crepes dulces a las crepes saladas" se representa como

prefiero	las	crepes	dulces	a	saladas
1	2	2	1	1	1

- Ignoramos el orden, lo cual implica ignorar contenido semántico. El documento "Prefiero las crepes saladas a las crepes dulces" se representa con el mismo vector.
- No es un problema si el objetivo es identificar *de qué habla* el documento y no *qué quiere decir*

Term-Document Matrix

- Organizamos los documentos en una matriz de términos X en el que celda (i, j) es número de veces que el token j aparece en el documento i.
- Términos en columnas definen *vocabulario* de un corpus
- Potencialmente muy grande y con muchos 0. Nuestro objetivo es reducirla un poco con preprocesado.
  1. Tokenización y normalización
  2. Eliminación de palabras vacías ("a", "de", "el", ...)
  3. Eliminación de palabras muy infrecuentes
  4. Uso de sinónimos (como WordNet)

Term frequency

- La distribución de uso de palabras es muy asimétrica
  - Unos pocos términos son muy frecuentes
  - Muchos términos son poco frecuentes
  - Relevancia no crece linealmente con frecuencia
- Podemos sustituir frequencia del término en cada documento por

$$\text{tf}_{ij} = (1 + \log_{10}(x_{ij}))$$

- Limita diferencias pequeñas en frecuencias en términos

Inverse document frequency

- También queremos una medida de frecuencia *entre* documentos
  - Palabras que aparece en menos documentos son más informativas
- Inversa de frecuencia de término en documentos

$$\text{idf}_j = \log_{10}\left(\frac{N}{\text{df}_j}\right)$$

  con N el número de documentos y df_j el número de documentos en los que aparece j
- Alto peso a palabras extrañas.

Ponderación TF-IDF

- Práctica común es usar como elementos en X

$$x_{ij} = \mathsf{tf}_{ij}\mathsf{idf}_j \tag{1}$$

- El peso crece con el número de ocurrencias en un documento y con la escasez en la colección.
- Representar documentos como vectores (cada término es una dimensión)
- Podemos calcular distancia entre documentos. Es mejor el ángulo (coseno) que la distancia euclidea, ya refleja *longitud* del documento.

Métodos de diccionario

- Es la alternativa más intuitiva. El analista define un diccionario de términos asociados a diferentes temas.

```
{
  "salud": ["hospital", "doctora", "enfermedad"],
  "economía":  ["inflación", "deuda", "crecimiento"],
  "defensa":  ["ejército", "guerra", "general"]
}
```

- Definimos la indicencia de cada tema en un documento sobre el total de palabras.
- Varios problemas
  - Refleja los supuestos del analista y no la estructura de los datos
  - Intensivo en trabajo
  - Cada corpus requiere un nuevo diccionario

Modelos probabilísticos de temas

- El modelo más popular es el Latent Dirichlet Allocation.
- Métodos no supervisado y generativo. Analiza los datos sin información sobre a qué tema pertenece cada documento. Producen:
  - Grupos de palabras que se refieren al mismo tema
  - Asocia temas a documentos
- Un "tema" es un conjunto de palabras que aparecen en el mismo contexto

Definir un tema

- El proceso que hace que determinadas palabras aparezcan en un mismo corpus
  - Si "guerra" o "ejército" aparecen en un documento, es probable que encontremos "tropas"
- La probabilidad de un término podemos descomponerla como

$$\Pr(\text{token} = \text{`manzana'}, \text{tema} = 1 | \theta, \beta) =$$
$$\Pr(\text{token} = \text{`manzana'} | \text{tema} = 1, \beta) \Pr(\text{tema} = 1 | \theta)$$

- Cada tema tiene una distribución sobre tokens
- Cada documento tiene una distribución sobre temas

El modelo generativo

1. Para cada tema, decidir qué temas son las más adecuados
2. Para cada documento
   - 2.1 Decidir qué proporciones de temas debe haber en el documento
   - 2.2 Para cada palabra
     - 2.2.1 Escoger a qué tema pertenece
     - 2.2.2 Dado ese tema, escoger la palabra más probable (generadas en el primer paso)

Estimación

- Si supiésemos a qué tema pertenece cada token estimar los parámetros sería calcular proporciones.
- Pero podemos pretender que lo sabemos (para una asignación) y calcular los parámetros. Después podemos tomar esos parámetros y calcular la distribución de términos.

Estimación

- Algoritmo EM tiene garantizada convergencia pero podemos suavizar los resultados asumiendo una distribución (Dirichlet) *a priori* que produce "mezclas" en varias dimensiones.
- Imponemos priors a nuestros parámetros

$$\Pr(\theta|\alpha) = Dirichlet(\alpha)$$
$$\Pr(\beta|\eta) = Dirichlet(\eta)$$

- Y simulamos usando Monte Carlo

Selección de número de temas

- La estimación del modelo toma el número de temas como dado
- Igual que como en k-means la mejor estrategia es comparar el funcionamiento del modelo para diferente número de temas. Varias alternativas:
  - Perplejidad y verosimilitud: capacidad de predecir datos (comparar distribución de frecuencia de palabras empírica y esperada).
  - Evaluación humana (cohesión de temas e intrusión de términos)