PYTHON PARA LINGÜISTAS

ANÁLISIS DE SENTIMIENTO

ALEJANDRO ARIZA

CENTRE DE LLENGUATGE I COMPUTACIÓ

UNIVERSITAT DE BARCELONA

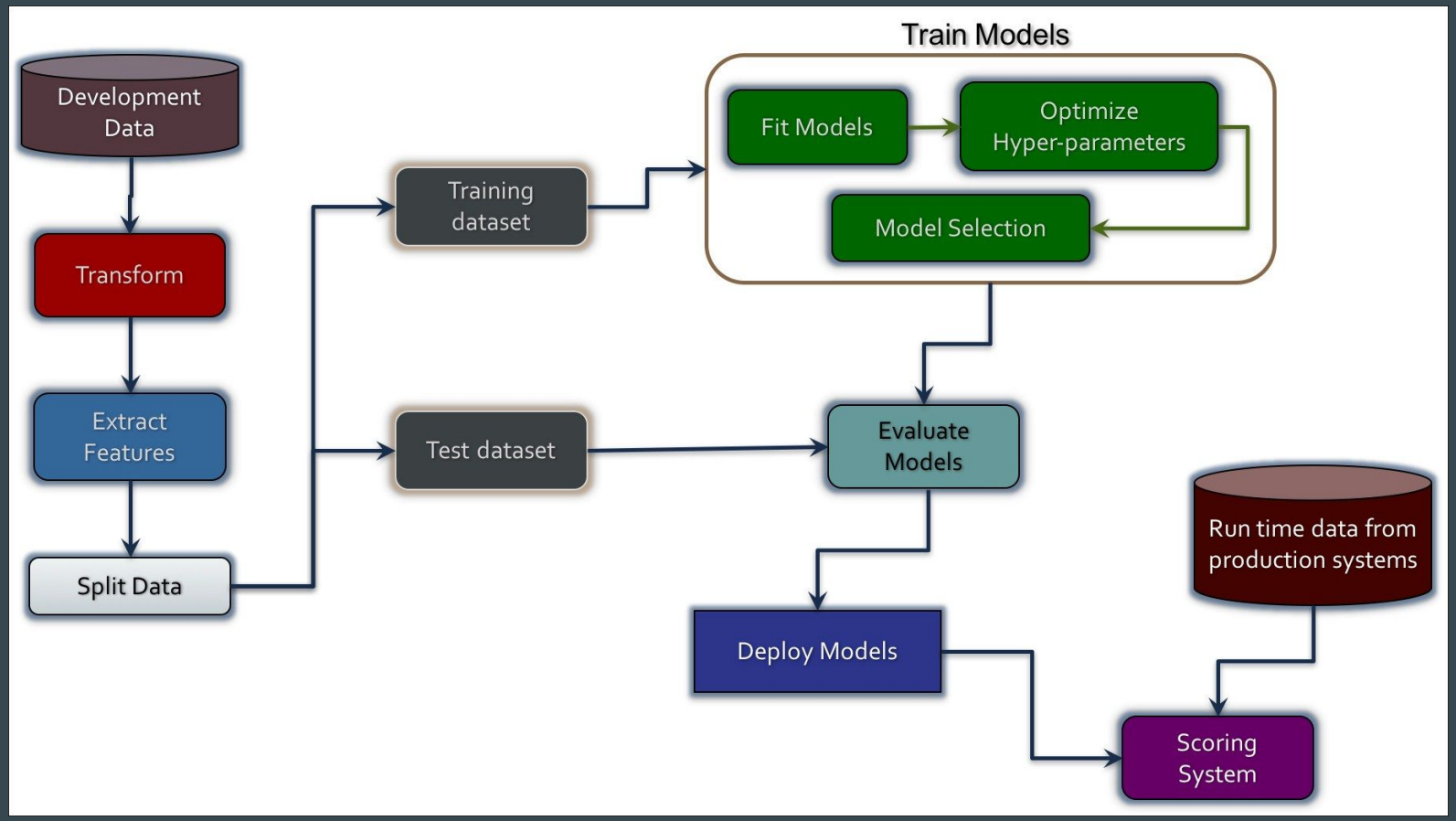
¿QUÉ HEMOS VISTO?

- Fundamentos de programación:
  - Tipos de datos y variables
  - Funciones y métodos
  - Estructuras de decisión
  - Bucles
- Preprocesado básico de corpus:
  - Limpieza
  - Tokenización y segmentación de frases
  - Estadísticas simples de un corpus y n-gramas
  - Etiquetado POS. Aprendizaje automático supervisado.
  - Análisis sintáctico: Constituyentes y dependencias.

STEMMING

- El stemming, con el mismo objetivo que la lematización, busca obtener la raíz de una palabra.
- Cuando trabajamos con modelos estadísticos, es posible que queramos utilizar únicamente la raíz de la palabra dado que la terminación no nos aporta información extra para la tarea a realizar.
- Un algoritmo que se suele utilizar para stemming es el algoritmo de Porter que nos permite realizar la extracción del morfema de la palabra en cuestión. NLTK nos proporciona una implementación con el nombre de PorterStemmer.
- Ejemplo de stemming:
    - moving → mov, morning → morn, etc

APRENDIZAJE SUPERVISADO



PREPROCESSING

- Puntos a tener en cuenta:
  - Stopwords
  - Mayúsculas o minúsculas
  - Símbolos de puntuación
  - Acentuación o caracteres extraños
  - Tokenización y segmentación. N-gramas.
  - Hashtags, URLs, menciones, RTs, etc

EXTRACCIÓN DE FEATURES

- Frecuencias
  - Palabras, n-gramas, caracteres, etc.
- Representaciones vectoriales
  - Word2vec, GloVe, FastText, BERT, GPT, etc.
- Otras métricas:
  - ROUGE, BLEU, etc.
- Nota importante: si se combinan features hay que asegurarse que están codificados (si el modelo lo requiere) y, en tal caso, normalizados de igual forma.

PREPARACIÓN DE EXPERIMENTOS / MUESTREO

- Separación de datos en train-validation-test
- X → feature set, y → label set
- Técnicas estadísticas:
  - K-Fold / leave-one-out cross-validation
  - Repeated holdout
  - Bootstrapping

MODEL TRAINING AND EVALUATION

- Partición del dataset e.g. 80%-10%-10% (train-valid-test)
- X_train + y_train se usan para entrenar el modelo.
- X_valid + y_valid sirven para validar el modelo y seleccionar los hiper-parámetros que mejor funcionan para la tarea.
- X_test + y_test se utilizan para evaluar el modelo frente a datos que no ha visto (habilidad de generalización)

MODELO DE EJEMPLO: NAIVE BAYES

$$\text{loglikelihood} = \log\left(\frac{P(W_{pos})}{P(W_{neg})}\right)$$

$$P(W_{pos}) = \frac{freq_{pos} + 1}{N_{pos} + V}$$
$$P(W_{neg}) = \frac{freq_{neg} + 1}{N_{neg} + V}$$

$$p = logprior + \sum_{i}^{N}(loglikelihood_i)$$

$$logprior = log(D_{pos}) - log(D_{neg})$$

¡GRACIAS!
¿PREGUNTAS?