Aproximación práctica a la ciencia de los datos y Big Data

Mahout práctico

José Manuel Benítez Sánchez

Contenido

- Instalación de Mahout
- Ejemplos «in-memory»:
  - Ejemplo de clustering
  - Ejemplo de clasificación
- Ejemplos con MapReduce:
  - Ejemplo de clustering
  - Ejemplo de clasificación
- Clasificadores para «poker»

Instalación

- Verificar las dependencias:
  - Maven
  - Java
- Conectarse a http://apache.rediris.es/mahout y descargar la última versión
- Instalarla en local
- Configurar las variables de entorno

Acceso a la cuenta remota

- Acceder a la cuenta asignada en atlas.ugr.es
- Crear un directorio personal en HDFS
- Configurar variables de entorno:
  - JAVA_HOME
  - MAHOUT_HOME
  - HADOOP_HOME

Ejemplo de clustering

```
cd $MAHOUT_HOME

./examples/bin/cluster-reuters.sh
```

Comparar los resultados con distintos algoritmos y parámetros

Clasificación:_puntos con relleno

- Resolver el problema de detección de puntos con relleno
- Datos: usr/share/doc/mahout-doc-0.7+16/examples/src/main/resources/
- (Mantados de procesamiento en material sobre Mahout)
  - Preparación de datos
  - Construcción del clasificador
  - Evaluación del clasificador
  - Mejoras

Construyendo el clasificador

```
mahout trainlogistic --input donut.csv --output
./model --target color --categories 2 --
predictors x y --types numeric --features 20 --
passes 100 --rate 50
```

Clasificación: 20 newsgroups

```
cd $MAHOUT_HOME

./examples/bin/classify-20newsgroups.sh
```

Evaluar el rendimiento de distintos algoritmos.

Después, realizar la ejecución de las distintas etapas, paso a paso

Ejecución detallada

1. Crear directorio de trabajo
2. Decargar y desempaquetar datos
3. Convertir los datos en SequenceFiles
4. Preprocesar los datos para incluir frecuencias de términos
5. Dividir el conjunto de datos en: entrenamiento y prueba
6. Entrenar el clasificador
7. Evaluar el clasificador

1. Crear un directorio de trabajo

```
export WORK_DIR=/tmp/mahout-work-${USER}
mkdir -p ${WORK_DIR}
```

2. Descargar y desempaquetar datos

```
curl
http://people.csail.mit.edu/jrennie/20Newsgroups/2
0news-bydate.tar.gz -o ${WORK_DIR}/20news-
bydate.tar.gz

mkdir -p ${WORK_DIR}/20news-bydate

cd ${WORK_DIR}/20news-bydate && tar xzf ../20news-
bydate.tar.gz && cd .. && cd ..

mkdir ${WORK_DIR}/20news-all

cp -R ${WORK_DIR}/20news-bydate/*/*
${WORK_DIR}/20news-all
```

2.b Si se usa Hadoop

```
hadoop dfs -put ${WORK_DIR}/20news-all
${WORK_DIR}/20news-all
```

3. Convertir los datos en SequenceFile

```
mahout seqdirectory
        -i ${WORK_DIR}/20news-all
        -o ${WORK_DIR}/20news-seq
        -ow
```

4. Preprocesar los datos: frecuencias de términos

```
mahout seq2sparse
        -i ${WORK_DIR}/20news-seq
        -o ${WORK_DIR}/20news-vectors
        -lnorm
        -nv
        -wt tfidf
```

5. Dividir el conjunto de datos

```
mahout split
        -i ${WORK_DIR}/20news-vectors/tfidf-vectors
        --trainingOutput ${WORK_DIR}/20news-train-vectors
        --testOutput ${WORK_DIR}/20news-test-vectors
        --randomSelectionPct 40
        --overwrite --sequenceFiles -xm sequential
```

6. Entrenar el clasificador

```
mahout trainnb
        -i ${WORK_DIR}/20news-train-vectors
        -el
        -o ${WORK_DIR}/model
        -li ${WORK_DIR}/labelindex
        -ow
        -c
```

7. Evaluar el clasificador

```
mahout testnb
        -i ${WORK_DIR}/20news-test-vectors
        -m ${WORK_DIR}/model
        -l ${WORK_DIR}/labelindex
        -ow
        -o ${WORK_DIR}/20news-testing
        -c
```

Clasificación: poker

Diseñar y crear un clasificador para el problema de "poker".

Evaluar rendimiento y tiempos de respuesta con distintos algoritmos y parámetros.

Datos: /home/curso/poker