ESERCITAZIONE PIATTAFORMA WEKA

Croce Danilo

Web Mining & Retrieval 2015/2016

Outline

- Weka: a brief recap
  - ARFF Format
  - Performance measures
    - Confusion Matrix
    - Precision, Recall, F1, Accuracy
- Question Classification
  - Text Mining with Weka

Intro WEKA

- Collection of ML algorithms - open-source Java package
  - http://www.cs.waikato.ac.nz/ml/weka/
- Documentation
  - http://www.cs.waikato.ac.nz/ml/weka/index_documentation.html
- Schemes for classification include:
  - Decision trees, rule learner
  - Naive bayes
  - KNN
  - SVM
- For classification, Weka allows train/test split or Cross-fold validation

ARFF File

- Require declarations of @RELATION, @ATTRIBUTE and @DATA
  - `@RELATION` declaration associates a name with the dataset
    - `@RELATION <relation-name>`
  - `@ATTRIBUTE` declaration specifies the name and type of an attribute
    - `@ATTRIBUTE <attribute-name> <datatype>`
- Datatype can be numeric, nominal, string or date

```
@ATTRIBUTE sepallength NUMERIC
@ATTRIBUTE petalwidth NUMERIC
@ATTRIBUTE class {Setosa,Versicolor,Virginica}
```

- `@DATA` declaration is a single line denoting the start of the data segment

```
@DATA
1.4, 0.2, Setosa
1.4, ?, Versicolor
```

Performance measures

- Load IRIS dataset
  - http://www.cas.mcmaster.ca/~cs4tf3/iris.arff
- Execute a Decision Tree (J48) algorithm on the IRIS dataset, evaluating using:
  - Cross-validation
  - Percentage split

In output notice:

  - Confusion matrix
  - True positive, true negative, false positive, false negative
  - Precision, recall, f1-measure, accuracy
- Visualize the resulting decision tree

Question Classification

- Question classification consists in assigning a question to a class reflecting the intention of the question.

Example: “*What is the width of a football field*?” → `Number`

- This dataset contains data used in the work presented in [1], that also provides question class definitions, as well as the description of the training and testing sets.

[1] Xin Li, Dan Roth, Learning Question Classifiers. COLING'02, Aug., 2002.

The QC dataset

- A QC dataset is available at:

  http://cogcomp.cs.illinois.edu/Data/QA/QC/

- Train dataset: 5,452 questions
- Test dataset: 500 questions

- Two settings:
  - Coarse-Grained: 6 classes 🡸 We will focus on this setting
  - Fine-grained: 50 classes

- You will find two .arff file containing this dataset on the course page.

The QC dataset in arff

```
@RELATION coarse_qc_train

@ATTRIBUTE question STRING
@ATTRIBUTE __class__ {NUM,LOC,HUM,ENTY,DESC,ABBR}

@DATA
"How did serfdom develop in and then leave Russia ?","DESC"
"What films featured the character Popeye Doyle ?","ENTY"
"How can I find a list of celebrities ' real names ?","DESC"
"What fowl grabs the spotlight after the Chinese Year of the Monkey ?","ENTY"
"What is the full form of .com ?","ABBR”
...
```

Example

Class

- How can we apply a learning algorithm over a question as a string?

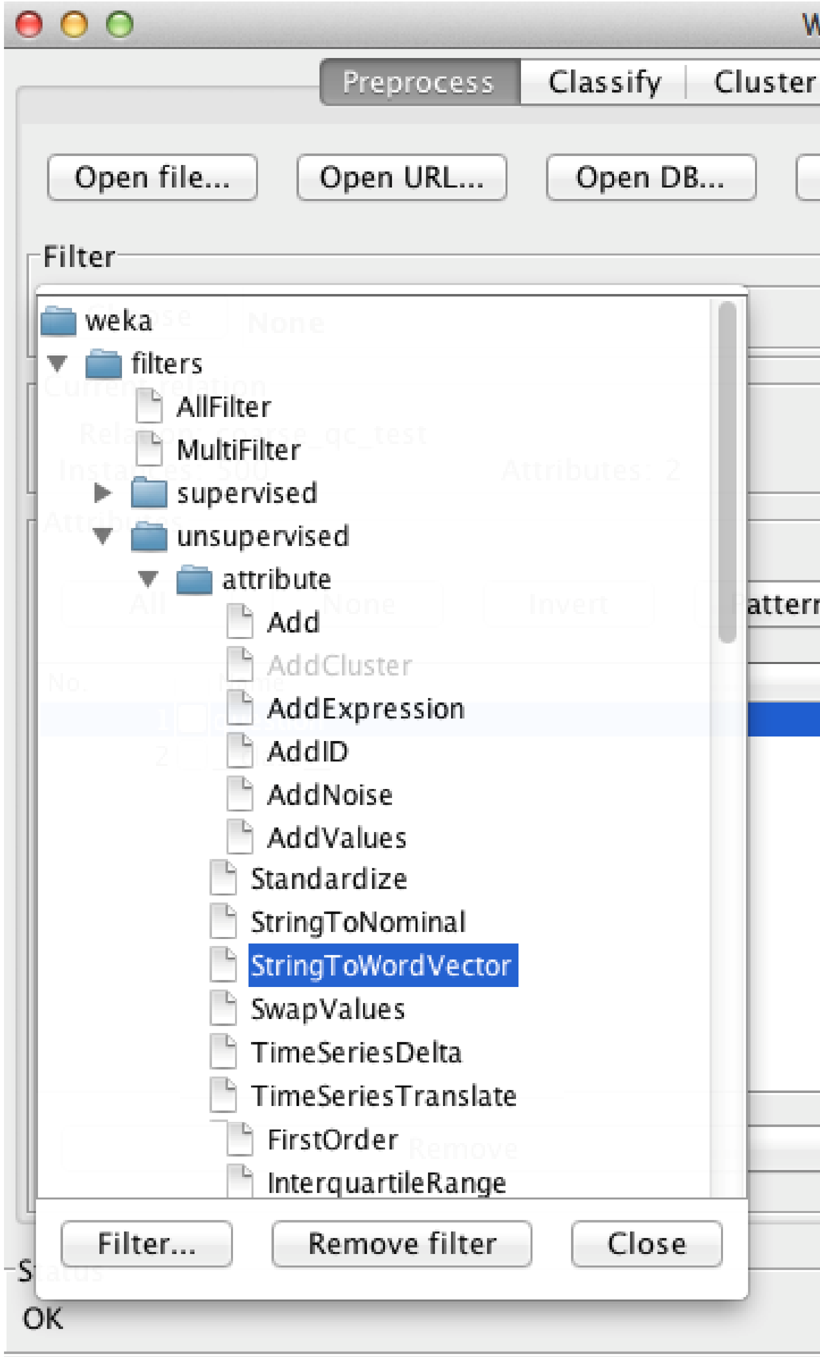
The `StringToWordVector` filter

- The filter

`weka.filters.unsupervised.attribute.StringToWordVector`

allows converting a dataset of strings into a dataset of vectors

  - The representation space has as many dimensions as words occurring in the dataset
  - A dimension of the vector will contain a non-zero element if the text contains the corresponding word



The `StringToWordVector` filter: usefull options

- `TFTransform/IDFTransform:` they allow estimating the term frequency (*tf*) and inverse document frequency (*idf*).
- `lowerCaseTokens`: If set then all the word tokens are converted to lower case before being added to the dictionary.
- `minTermFreq`: Sets the minimum term frequency: all words whose frequency is lower than `minTermFreq` are ignored.
- `stemmer`: The stemming algorithm to use on the words (e.g., "*argue*", "*argued*", "*argues*", "*arguing*", and "*argus*" reduce to the stem "*argu*").
- `tokenizer`: The tokenizing algorithm to use on the strings to split them into words.
- `useStoplist`: Ignores all the words that are on the stoplist, if set to true.
- `stopwords`: The file containing the stopwords (if this is a directory then the default ones are used).

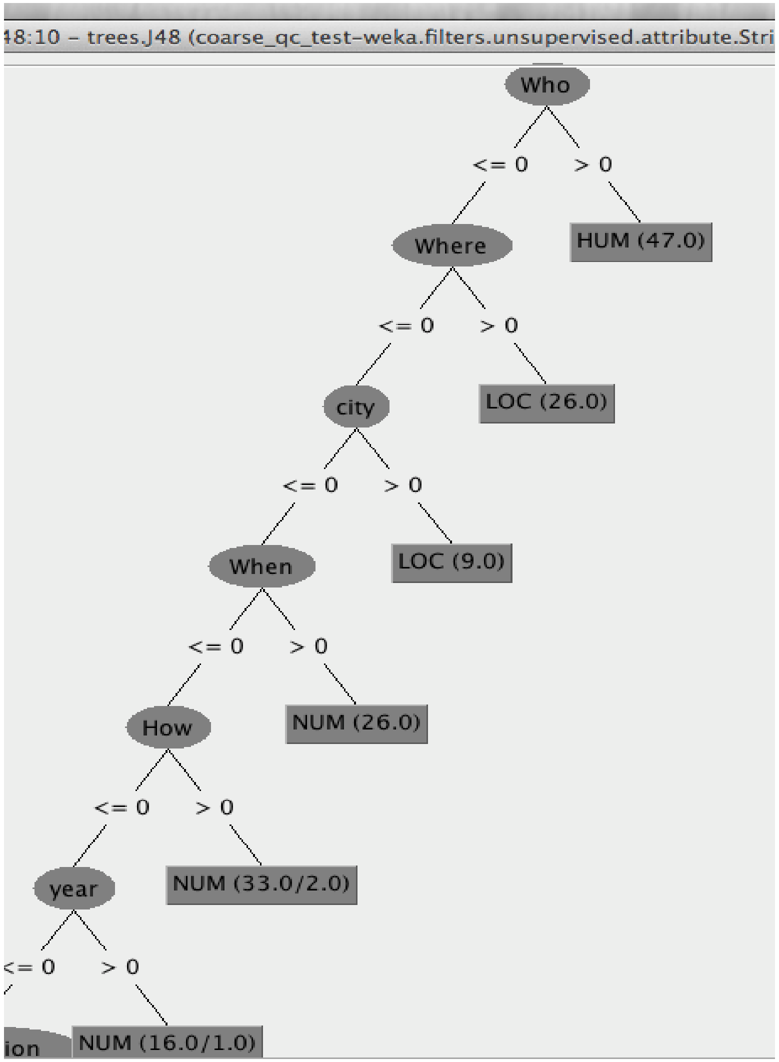
Acquiring a QC classifier

- Execute a Decision Tree (J48) algorithm on the IRIS dataset, evaluating using:
  - Cross-validation
  - Percentage split

In output notice:

  - Confusion matrix
  - True positive, true negative, false positive, false negative
  - Precision, recall, f1-measure, accuracy

- Visualize the resulting decision tree



48:10 – trees.J48 (coarse_qc_test-weka.filters.unsupervised.attribute.Stri
Who
<= 0
> 0
HUM (47.0)
Where
<= 0
> 0
LOC (26.0)
city
<= 0
> 0
LOC (9.0)
When
<= 0
> 0
NUM (26.0)
How
<= 0
> 0
NUM (33.0/2.0)
year
<= 0
> 0
ion
NUM (16.0/1.0)