CSC 5390/9010: Text Mining
Machine Learning with GATE

Dr. Paula Matuszek
Paula.Matuszek@villanova.edu
Paula.Matuszek@gmail.com
(610) 647-9789
In the last assignment we looked at how to set up GATE to do information extraction “by hand”.

- Gazeteers
- JAPE rules

We modified the system to add UPenn and Villanova to universities to be extracted

Not hard for two entities, but this can get tricky. Is there a better way?
We would like:
• give GATE examples of universities
• let it figure out for itself how to identify them

This is basically a classification problem: given an entity, is it a university?

This sounds familiar!

The classification methods we looked at in NLTK can be considered as forms of supervised machine learning.
Machine Learning in GATE

• GATE has machine learning processing resources: CREOLE Learning plugin

• Same classification algorithms we saw in NLTK

• But both features to be used and class to be learned can be richer, using PRs like ANNIE
Machine Learning

We have data items comprising labels and features

E.g. an instance of “cat” has features “whiskers=1”, “fur=1”. A “stone” has “whiskers=0” and “fur=0”

Machine learning algorithm learns a relationship between the features and the labels

E.g. “if whiskers=1 then cat”

This is used to label new data

We have a new instance with features “whiskers=1” and “fur=1”--is it a cat or not???
ML in Information

We have annotations (classes)
We have features (words, context, word features etc.)
Can we learn how features match classes using ML?
Once obtained, the ML representation can do our annotation for us based on features in the text
  Pre-annotation
  Automated systems
Possibly good alternative to knowledge engineering approaches
  No need to write the rules
However, need to prepare training data
Central to ML work is evaluation

Need to try different methods, different parameters, to obtain good result

Precision: How many of the annotations we identified are correct?
Recall: How many of the annotations we should have identified did we?

F-Score:
\[ F = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \]

Testing requires an unseen test set

Hold out a test set
  - Simple approach but data may be scarce

Cross-validation
  - split training data into e.g. 10 sections
  - Take turns to use each “fold” as a test set
  - Average score across the 10
More on SVMs

• Primary Machine Learning engine in GATE is the Support Vector Machine
  • Handles very large feature sets well
  • Has been shown empirically to be effective in a variety of unstructured text learning tasks.

• We covered this earlier; we will look at using them within GATE
Basic Idea Underlying SVMs

• Find a line, or a plane, or a hyperplane, that separates our classes cleanly.
  • This is the same concept as we have seen in regression.
• By finding the greatest margin separating them
The maximum margin linear classifier is the linear classifier with the maximum margin. Called Linear Support Vector Machine (SVM).
Maximum Margin

- denotes +1
- denotes -1

Support Vectors are those datapoints that the margin pushes up against

The maximum margin linear classifier is the linear classifier with the, um, maximum margin. Called Linear Support Vector Machine (SVM)
Slack Variables and Cost

• In order to find a soft margin, we allow \textit{slack variables}, which measure the degree of misclassification.

• Takes into account the number of misclassified instances and the distance from the margin

• We then modify this by a cost (C) for these misclassified instances.

• High cost -- narrow margins, won’t generalize well

• Low cost -- broader margins but misclassify more data.

• How much we want it to cost to misclassify instances depends on our domain -- what we are trying to do
What If We Want Three Classes?

• Suppose our task involves more than two classes: universities, cities, businesses?

• Reduce multiple class problem to multiple binary class problems.
  • one-versus-all, one-vs-others
  • one-versus-one, one-vs-another

• GATE will do this automatically if there are more than two classes.
Doing this in GATE

- GATE has a Learning plugin available in CREOLE. The PR available from it is Batch Learning
  - Newest machine learning PR
  - focused on chunk recognition, text classification, relation extraction.
  - Primary algorithm is an SVM
  - Can also interface to WEKA for Naive Bayes, KNN and decision trees
- There is also an older Machine_Learning plugin, which contains wrappers for several different external learning systems.
Using Batch Learning PR

- The batch learning PR is basically learning new annotations (the class) from previous annotations (the features)

- Need three things:
  - annotated training documents
  - possibly preprocessed to get annotations we want to learn from
  - an XML configuration file (which is external to the IDE)

- The PR treats the parent directory of the config file as its working directory.
ML applications in GATE

- Batch Learning PR
  - Evaluation
  - Training
  - Application
- Runs after all other PRs – must be last PR
- Configured via xml file
- A single directory holds generated features, models, and config file
Instances, attributes, classes

California Governor Arnold Schwarzenegger proposes deep cuts.

**Instances:** Any annotation
- Tokens are often convenient

**Attributes:** Any annotation feature relative to instances
- Token.String
- Token.category (POS)
- Sentence.length

**Class:** The thing we want to learn
- A feature on an annotation

- **Entity.type=Location**
- **Entity.type=Person**
Instances

- Instances are cases that may be learned
- Every instance is a decision for the ML algorithm to make
- To which class does this instance belong?
Attributes

Attributes are pieces of information about instances

They are sometimes called “features” in machine learning literature
Classes

The class is what we want to learn

For example, if we want to find person names, for every instance, the question is, is this a person name?

The classes are “yes” and “no”

Sometimes there are many classes, for example we may want to learn entity types

For every instance, the question is, which of a predetermined entity type set does this belong to?
Training

Training involves presenting data to the ML algorithm from which it creates a model.

The training data (instances) have been annotated with class annotations as well as attributes.

Models are representations of decision-making processes that allow the machine learner to decide what class the instance has based on the attributes of the instance.
Application

When the machine learner is applied, it creates new class annotations on data using the model.

The corpus it is applied to must contain the required attribute annotations.

The machine learner will work best if the application data is similar to the training data.
Evaluation

We want to know how good our machine learner is before we use it for a real task

Therefore we apply it to some data for which we already have class annotations

The “right answers”, sometimes called “gold standard”

If the machine learner creates the same annotations as the gold standard, then we know it is performing well

The test corpus must not be the same corpus as you trained on

This would give the machine learner an advantage, and would give a false idea of how good it is

GATE's ML PR has a built-in evaluation mode that splits the corpus into training and test sets and cross-validates them
Summary

- GATE supports a very rich set of classifier-type machine learning capabilities
  - Current: the Learning plugin has the Batch Learner
  - Older: Machine_Learning plugin has wrappers for various ML systems
- Both class to be learned and features to learn from are document annotations
- Same algorithms as we saw in NLTK, but use can be very different
  - classifying chunks of text within documents
  - alternative to engineering JAPE rules by hand
  - can also use to classify entire document
For lab

- For lab we are going to work through another GATE tutorial.

- Download the materials at
  - [http://gate.ac.uk/wiki/TrainingCourseJune2013/](http://gate.ac.uk/wiki/TrainingCourseJune2013/)
  - Track 3, Wednesday, Module 11. Slides, exercise sheet, hands-on material.
  - We don’t need the supplementary materials today