

University of Science and Technology of Hanoi



Text Classification Using Decision Tree and Maximum Entropy

Student: **Bui Dinh Duong**

Hanoi, September 20th, 2013

Organization

- Introduction
- Objectives
- Text Classification Overview
- Decision Tree
- Maximum Entropy
- Experiment
- Conclusion

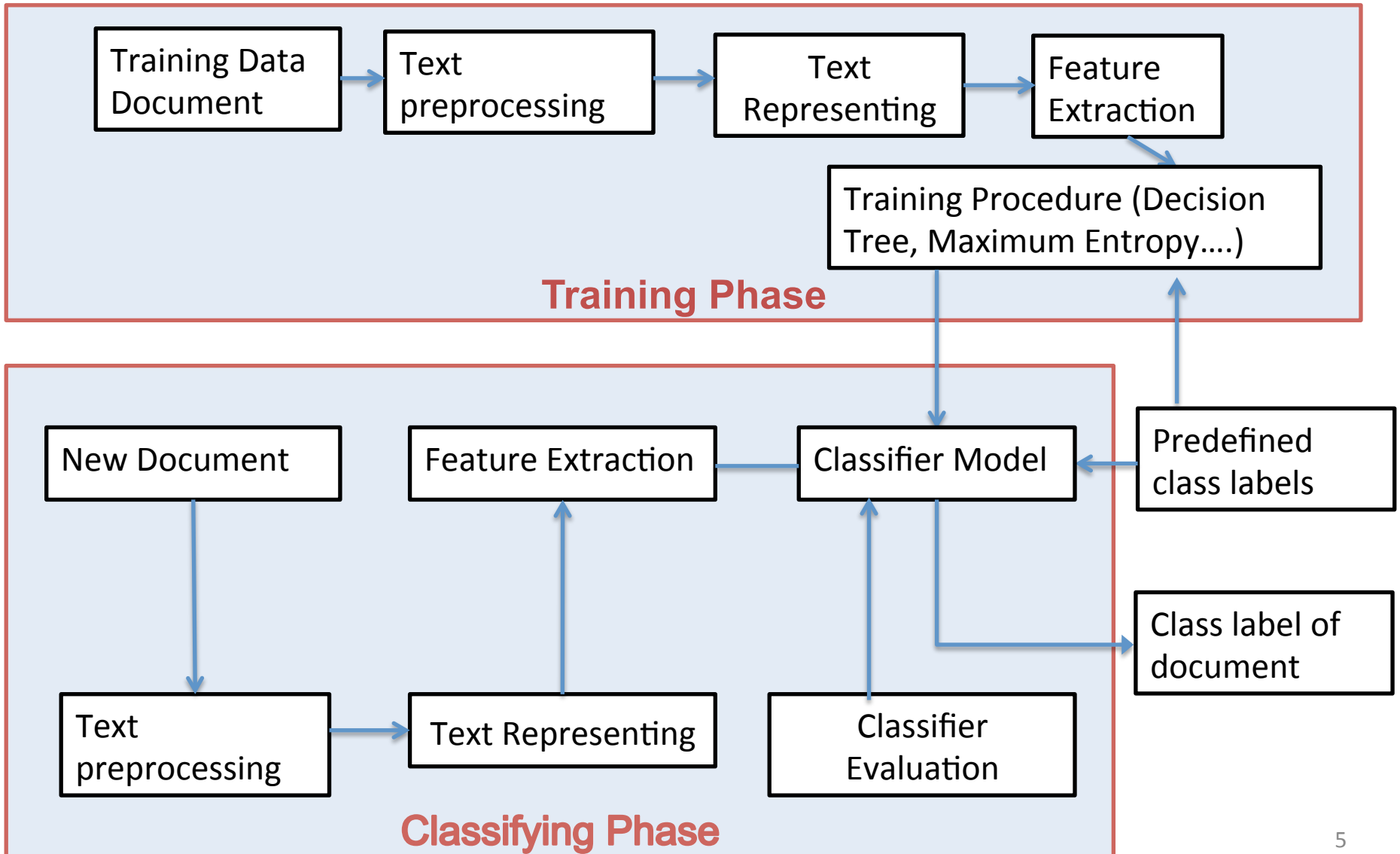
Introduction

- Text classification
 - Assign a document to one or more predefined classes
- Applications
 - E-mail spam filtering
 - Categorize newspaper articles into topics
 - Organize Web pages into hierarchical categories
 - Language identification
- Methods
 - Naive Bayes
 - Maximum Entropy
 - Decision Tree
 - Support Vector Machine (SVM)

Objectives

- Study the stages of text classification
- Study Decision Tree method
- Study Maximum Entropy method
- Do experiment in text classification using Weka

Text Classification



Text Preprocessing

- What is the objective?
 - Reduce the size of data
 - Get only things we need
- How to do?
 - Convert document to lower case
 - Remove words that rarely occur in the document
 - Remove special character
 - Remove stop-words (words are not used to classify)
 - Remove suffix, prefix of word to get the root word (“clusters”, “clustering”, “clustered” => cluster)

Text Representing

➤ What is the objective?

- Represent text data in a suitable model to process

➤ How to do?

- Vector Space Model (most popular method)
 - Each document is represented as a vector of **word weighting**

For example: “**The brown fox jumps over the lazy dog**”

a	an	...brown,...	dog	...	fox	jump	lazi	over	the
		↓	↓		↓	↓	↓	↓	↓
(0, 0, ..., 0,	1,	0, ..., 0,	1,	0, ..., 0,	1,	0, ..., 0,	1,	0, ..., 0,	1,
2, 0, ..)									

Feature Extraction

➤ Word Weighting

- *Word frequency weighting* and *TF*IDF weighting*: the number of time that a word appears in a document
- Three values are used to calculate the weighting:
 - **Term frequency**: the number of time a word appears in a document
 - **Collection frequency**: the number of time a word appears in document collection (whole dataset)
 - **Document frequency**: the number of document contains a word

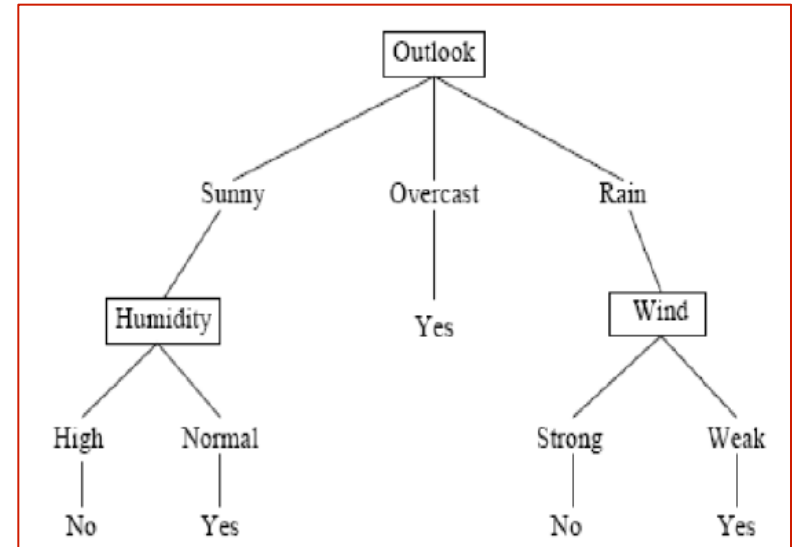
=> Features: words have highest Word weighting

Classifier Evaluation

- What is the objective?
 - Evaluate quality of the model and its accuracy to know if we can use this model or not
- How to do?
 - **Accuracy**: the proportion of correctly classified objects
 - **Error**: the proportion of incorrectly classified objects
 - **Precision**: the proportion of selected items that the system got right
 - **Recall**: the proportion of the target items that the system selected
 - **Fallout**: the proportion of no targeted items that were mistakenly selected
 - **F-measure**: Precision and Recall are combined

Decision Tree

- The first node is root node
- Internal nodes are attribute tests
- Leaf nodes are class label
- Many algorithms ID3, C4.5, CART, CHAID, MARS in decision tree
- ID3 uses Entropy and Information Gain
- Pruning
 - The pruning step is to avoid **over-fitting**
- Cross-validation
 - To maximize the accurate classification of classifier tree model



Decision Tree (ID3)

➤ Entropy

- Entropy is the indicator of how much information inside a data set

$$\textit{Entropy}(S) = \sum_{i=1}^C -p_i \log_2 p_i$$

Where:

- S: the set of training data
- C: the number of class labels
- p_i : the rate of elements belong to class C_i

Decision Tree (ID3)

➤ Information Gain

- Information gain is the measures of reducing entropy in S by an attribute in S

$$\text{Gain}(S, A) = \text{Entropy}(S) - \sum_{v \in \text{Value}(A)} \frac{|S_v|}{S} \text{Entropy}(S_v)$$

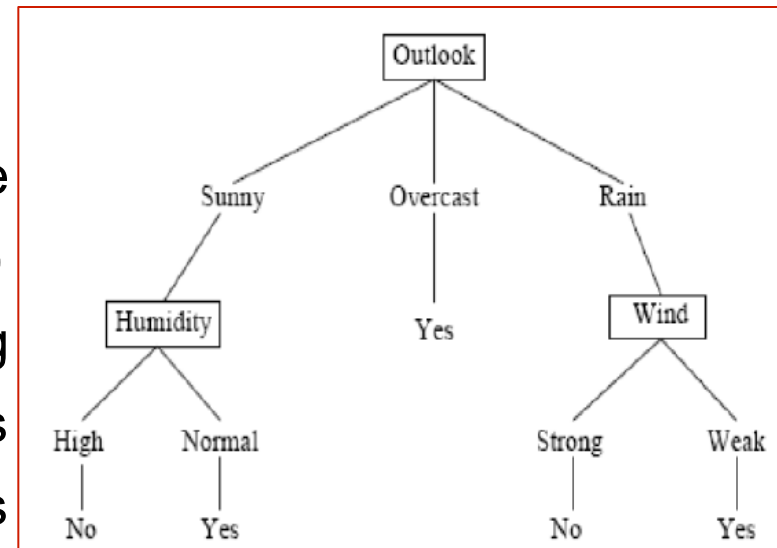
Where:

- Value (A): set of A values
- S_v : subset of S

Decision Tree (ID3)

➤ General steps of ID3 Algorithm

1. From the dataset S , calculate the entropy of every attribute (feature)
2. Split the set S into subsets using the attribute for which entropy is minimum (or information gain is maximum)
3. Expanding decision tree by adding a node containing that attribute
4. Recursive on subsets using remaining attributes



Maximum Entropy

➤ Main idea

- Satisfy constraints
- Probability distribution of model which is most uniform

➤ What is the constraint?

- Constraint : If a document contains the word “professor”, it has a 40% chance of probability distribution in faculty class

$$f_i(\vec{x}_j, c) = \begin{cases} 1, & \text{if } w_{ij} > 0 \text{ and } c = 1 \\ 0, & \text{otherwise} \end{cases}$$

w_{ij} is the word weighting of word i in document j

Maximum Entropy

➤ Log-linear Model

- Use to classify document in Maximum Entropy

$$p(\vec{x}, c) = \frac{1}{Z} \prod_{i=1}^K \alpha_i^{f_i(\vec{x}, c)}$$

Where :

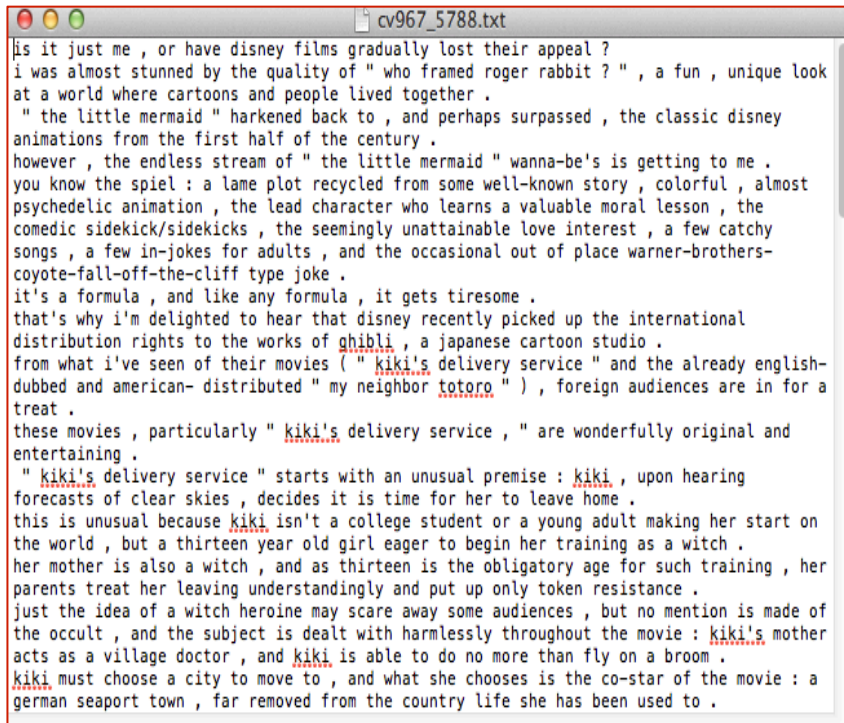
- K: the number of constraints
 - Z: a constant
 - α_i : the weight of f_i
- Compute $p(\vec{x}_{new}, 1)$ and $p(\vec{x}_{new}, 0)$.
New document belong to class which has higher probability

Maximum Entropy

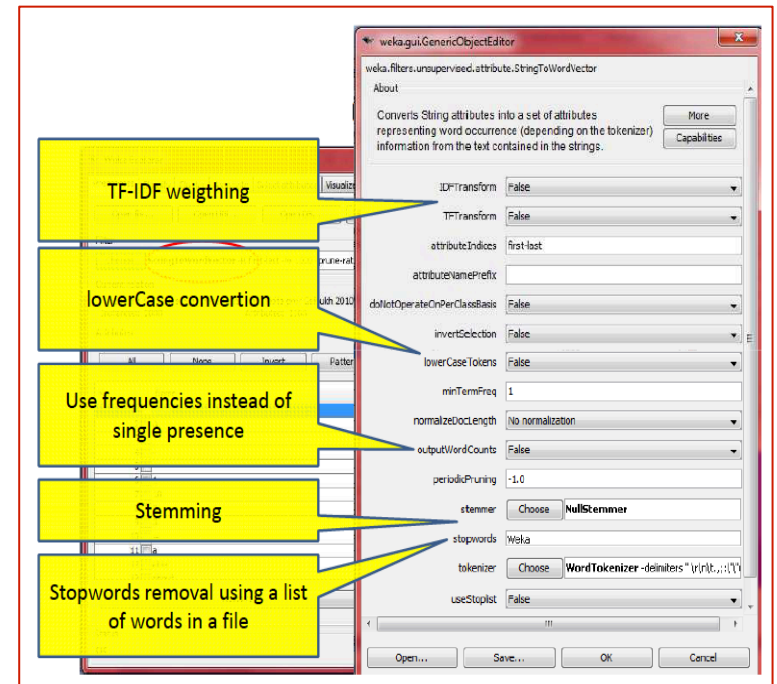
- Generalized iterative scaling (GIS)
 - Use to find α_i in the Log-linear Model
 - GIS find probability distribution which has maximum entropy of Log-linear Model

Experiment

- Dataset : 1000 negative movie reviews and 1000 positive movie reviews
- Text Preprocessing



is it just me, or have disney films gradually lost their appeal ?
i was almost stunned by the quality of " who framed roger rabbit ? ", a fun , unique look at a world where cartoons and people lived together .
" the little mermaid " harkened back to , and perhaps surpassed , the classic disney animations from the first half of the century .
however , the endless stream of " the little mermaid " wanna-be's is getting to me .
you know the spiel : a lame plot recycled from some well-known story , colorful , almost psychedelic animation , the lead character who learns a valuable moral lesson , the comedic sidekick/sidekicks , the seemingly unattainable love interest , a few catchy songs , a few in-jokes for adults , and the occasional out of place warner-brothers-coyote-fall-off-the-cliff type joke .
it's a formula , and like any formula , it gets tiresome .
that's why i'm delighted to hear that disney recently picked up the international distribution rights to the works of ghibli , a japanese cartoon studio .
from what i've seen of their movies (" kiki's delivery service " and the already english-dubbed and american- distributed " my neighbor totoro ") , foreign audiences are in for a treat .
these movies , particularly " kiki's delivery service , " are wonderfully original and entertaining .
" kiki's delivery service " starts with an unusual premise : kiki , upon hearing forecasts of clear skies , decides it is time for her to leave home .
this is unusual because kiki isn't a college student or a young adult making her start on the world , but a thirteen year old girl eager to begin her training as a witch .
her mother is also a witch , and as thirteen is the obligatory age for such training , her parents treat her leaving understandingly and put up only token resistance .
just the idea of a witch heroine may scare away some audiences , but no mention is made of the occult , and the subject is dealt with harmlessly throughout the movie : kiki's mother acts as a village doctor , and kiki is able to do no more than fly on a broom .
kiki must choose a city to move to , and what she chooses is the co-star of the movie : a german seaport town , far removed from the country life she has been used to .



Weka GUI Object Editor

weka.filters.unsupervised.attribute.StringToWordVector

About

Converts String attributes into a set of attributes representing word occurrence (depending on the tokenizer) information from the text contained in the strings.

More

Capabilities

Visualize

IDF Transform: False

TF Transform: False

attributeIndices: first-last

attributeNamePrefix:

dotListOperatorOnPerClassBasis: False

invertSelection: False

lowerCaseTokens: False

minTermFreq: 1

normalizedDocLength: No normalization

outputWordCounts: False

periodicPruning: -1.0

stemmer: Choose NullStemmer

stopwords: Weka

tokenizer: Choose WordTokenizer-delimiters "\r\n\t,;:!\",{}
\"

useStoplist: False

Open... Save... OK Cancel

TF-IDF weighing

lowerCase conversion

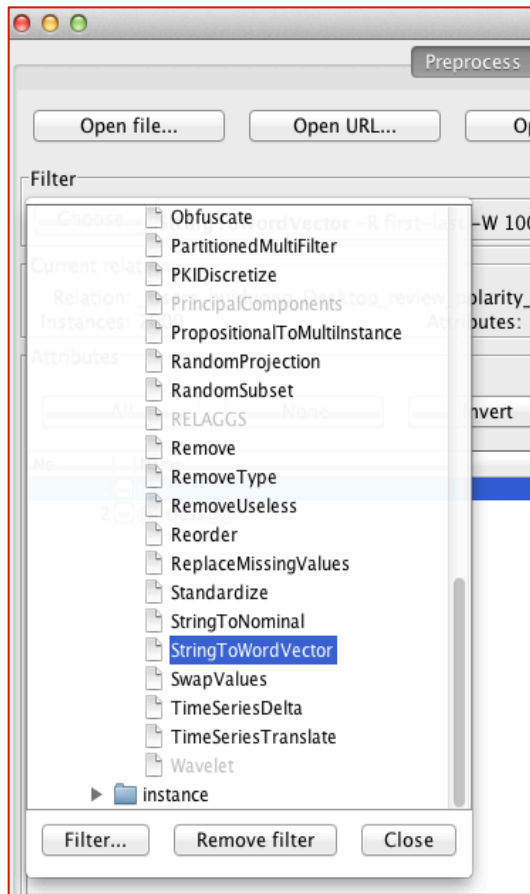
Use frequencies instead of single presence

Stemming

Stopwords removal using a list of words in a file

Experiment

➤ Text representing



review_trainingdata.arff — Edited

```
@attribute wars {0,1}
@attribute wedding {0,1}
@attribute wonderful {0,1}
@attribute wonderfully {0,1}
@attribute woody {0,1}

@data
{1 1,3 1,6 1,8 1,15 1,18 1,20 1,21 1,24 1,28 1,33 1,34 1,35 1,39 1,42 1,43 1,50 1,54 1,59 1,62 1,65 1,67 1,71 1,72
1,77 1,79 1,82 1,83 1,95 1,96 1,98 1,106 1,109 1,118 1,119 1,123 1,136 1,137 1,140 1,158 1,161 1,164 1,183 1,184 1,186
1,189 1,193 1,194 1,196 1,200 1,203 1,205 1,206 1,209 1,212 1,237 1,241 1,243 1,247 1,250 1,251 1,281 1,282 1,289
1,291 1,292 1,294 1,298 1,300 1,305 1,314 1,319 1,325 1,327 1,332 1,334 1,335 1,337 1,338 1,342 1,344 1,345 1,349
1,352 1,357 1,364 1,366 1,369 1,370 1,376 1,377 1,381 1,383 1,391 1,395 1,402 1,403 1,410 1,419 1,423 1,426 1,427
1,441 1,445 1,450 1,453 1,477 1,479 1,482 1,484 1,491 1,495 1,503 1,505 1,506 1,507 1,520 1,521 1,529 1,533 1,535
1,537 1,543 1,544 1,550 1,551 1,556 1,558 1,565 1,569 1,574 1,577 1,581 1,586 1,587 1,588 1,592 1,593 1,595 1,597
1,599 1,604 1,608 1,611 1,612 1,613 1,616 1,621 1,626 1,640 1,643 1,644 1,659 1,663 1,664 1,665 1,681 1,693 1,708
1,711 1,715 1,723 1,736 1,737 1,739 1,740 1,741 1,743 1,760 1,762 1,763 1,768 1,769 1,778 1,781 1,785 1,801 1,802
1,806 1,811 1,812 1,814 1,822 1,827 1,831 1,833 1,842 1,849 1,850 1,854 1,855 1,856 1,857 1,858 1,860 1,865 1,872
1,877 1,882 1,883 1,895 1,900 1,907 1,908 1,910 1,919 1,921 1,922 1,931 1,939 1,942 1,943 1,944 1,946 1,950 1,951
1,952 1,953 1,955 1,956 1,958 1,970 1,982 1,986 1,988 1,996 1,1000 1,1002 1,1016 1,1051 1,1098 1,1101 1}
```

Feature and word weighting

Experiment (result)

```

worst = 1
  bring = 0
    tom = 0
      details = 0: neg
      details = 1
        -- = 0: pos
        -- = 1: neg
    tom = 1
      come = 0: neg
      come = 1: pos
  bring = 1
    see = 0: neg
    see = 1
      usually = 0
        america = 0: pos
        america = 1: neg
      usually = 1: neg
wonderfully = 1
  red = 0: pos
  red = 1: neg
stupid = 1
  bob = 0
    into = 0
      perfect = 0
        certainly = 0: neg
        certainly = 1
          - = 0: pos
          - = 1: neg
      perfect = 1: pos
    into = 1: neg
  bob = 1
    10 = 0: pos
    10 = 1: neg

```

Correctly Classified Instances	247	61.75	%
Incorrectly Classified Instances	153	38.25	%
Kappa statistic	0.235		
Mean absolute error	0.3825		
Root mean squared error	0.6185		
Relative absolute error	76.5	%	
Root relative squared error	123.6932	%	
Total Number of Instances	400		

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	0.61	0.375	0.619	0.61	0.615	0.618	neg
	0.625	0.39	0.616	0.625	0.62	0.618	pos
Weighted Avg.	0.618	0.383	0.618	0.618	0.617	0.618	

Result of classifying phase
(training data 66%, test 34%)

Conclusion

➤ Achievements

- Understand the stages of text classification
- Gather two methods of text classification:
 - Decision Tree method
 - Maximum Entropy method

➤ Future works

- Continue researching methods of text classification
- Program decision tree method to classify document

Reference

- Christopher D. Manning, Hinrich Schütze, “**Foundations of Statistical Natural Language Processing**”
- Kamal Nigam, John Lafferty, Andrew McCallum, “**Using Maximum Entropy for Text Classification**”, In IJCAI-99 Workshop on Machine Learning for Information Filtering
- Kostas Fragos, Yannis Maistros, Christos Skourlas, “**A Weighted Maximum Entropy Language Model for Text Classification**”, NLUCS 2005: p.55-67
- Tom M. Mitchell, “**Machine learning**”, Published by McGraw-Hill, Maidenhead, U.K., International Student Edition, 1997. ISBN: 0-07-115467-1

Thank You!