Processing of large document collections

Part 3 (Evaluation of text classifiers, term selection)

Helena Ahonen-Myka

Spring 2006

Evaluation of text classifiers

- evaluation of document classifiers is typically conducted experimentally, rather than analytically
- reason: in order to evaluate a system analytically, we would need a formal specification of the problem that the system is trying to solve
- text categorization is non-formalisable

2

Evaluation

- the experimental evaluation of a classifier usually measures its effectiveness (rather than its efficiency)
  - effectiveness= ability to take the right classification decisions
  - efficiency= time and space requirements

3

Evaluation

- after a classifier is constructed using a training set, the effectiveness is evaluated using a test set
- the following counts are computed for each category i:
  - TP_i: true positives
  - FP_i: false positives
  - TN_i: true negatives
  - FN_i: false negatives

4

Evaluation

- TP_i: true positives w.r.t. category c_i
  - the set of documents that both the classifier and the previous judgments (as recorded in the test set) classify under c_i
- FP_i: false positives w.r.t. category c_i
  - the set of documents that the classifier classifies under c_i, but the test set indicates that they do not belong to c_i

5

Evaluation

- TN_i: true negatives w.r.t. c_i
  - both the classifier and the test set agree that the documents in TN_i do not belong to c_i
- FN_i: false negatives w.r.t. c_i
  - the classifier do not classify the documents in FN_i under c_i, but the test set indicates that they should be classified under c_i

6

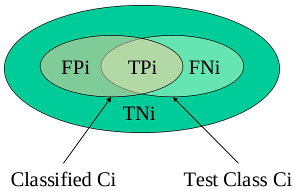
Evaluation measures

- Precision wrt c_i

$$\pi_i = \frac{TP_i}{TP_i + FP_i}$$

- Recall wrt c_i

$$\rho_i = \frac{TP_i}{TP_i + FN_i}$$



Evaluation measures

- for obtaining estimates for precision and recall in the collection as a whole (= all categories), two different methods may be adopted:
  - microaveraging
    - counts for true positives, false positives and false negatives for all categories are first summed up
    - precision and recall are calculated using the global values
  - macroaveraging
    - average of precision (recall) for individual categories

Evaluation measures

- microaveraging and macroaveraging may give quite different results, if the different categories are of very different size
  - e.g. the ability of a classifier to behave well also on small categories (i.e. categories with few positive training instances) will be emphasized by macroaveraging
- choice depends on the application

Combined effectiveness measures

- neither precision nor recall makes sense in isolation of each other
- the trivial acceptor (each document is classified under each category) has a recall = 1
  - in this case, precision would usually be very low
- higher levels of precision may be obtained at the price of lower values of recall

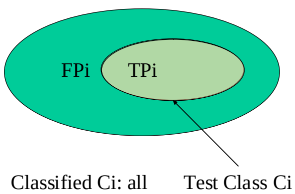
Trivial acceptor

- Precision wrt c_i

$$\pi_i = \frac{TP_i}{TP_i + FP_i}$$

- Recall wrt c_i

$$\rho_i = \frac{TP_i}{TP_i + FN_i}$$



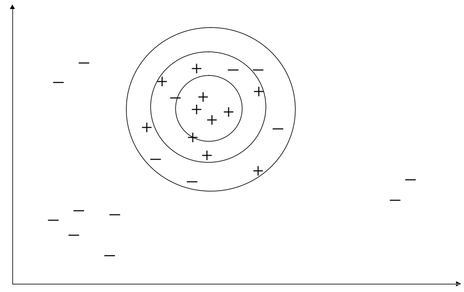
Combined effectiveness measures

- a classifier should be evaluated by means of a measure which combines recall and precision
- some combined measures:
  - 11-point average precision
  - the breakeven point
  - F1 measure

11-point average precision

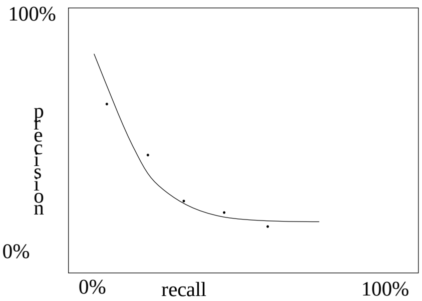
- in constructing the classifier, the threshold is repeatedly tuned so as to allow recall (for the category) to take up values 0.0, 0.1., ..., 0.9, 1.0.
- precision (for the category) is computed for these 11 different values of precision, and averaged over the 11 resulting values

13



14

Recall-precision curve



15

Breakeven point

- process analoguous to the one used for 11-point average precision
  - precision as a function of recall is computed by repeatedly varying the thresholds
- breakeven point is the value where precision equals recall

16

F_1 measure

- F_1 measure is defined as:

$$F_1 = \frac{2\pi\rho}{\pi + \rho}$$

- for the trivial acceptor, $\pi \to 0$ and $\rho = 1$, $F_1 \to 0$

17

Effectiveness

- once an effectiveness measure is chosen, a classifier can be tuned (e.g. thresholds and other parameters can be set) so that the resulting effectiveness is the best achievable by that classifier

18

Evaluation measures

- efficiency (= time and space requirements)
  - seldom used, although important for real-life applications
  - difficult to compare systems: environment parameters change
  - two parts
    - training efficiency = average time it takes to build a classifier for a category from a training set
    - classification efficiency = average time it takes to classify a new document under a category

19

Conducting experiments

- in general, different sets of experiments may be used for cross-classifier comparison only if the experiments have been performed
  - on exactly the same collection (same documents and same categories)
  - with the same split between training set and test set
  - with the same evaluation measure

20

Term selection

- a large document collection may contain millions of words -> document vectors would contain millions of dimensions
  - many algorithms cannot handle high dimensionality of the term space (= large number of terms)
  - very specific terms may lead to overfitting: the classifier can classify the documents in the training data well but fails often with unseen documents

21

Term selection

- usually only a part of terms is used
- how to select terms that are used?
  - term selection (often called feature selection or dimensionality reduction) methods

22

Term selection

- goal: select terms that yield the highest effectiveness in the given application
- wrapper approach
  - a candidate set of terms is found and tested with the application
  - iteration: based on the test results, the set of terms is modified and tested again until the set is optimal
- filtering approach
  - keep the terms that receive the highest score according to a function that measures the ”importance” of the term for the task

23

Term selection

- many functions available
  - document frequency: keep the high frequency terms
    - stopwords have been already removed
    - 50% of the words occur only once in the document collection
    - e.g. remove all terms occurring in at most 3 documents

24

Term selection functions: document frequency

- document frequency is the number of documents in which a term occurs
- in our sample, the ranking of terms:
  - 9 current
  - 7 project
  - 4 environment
  - 3 nuclear
  - 2 application
  - 2 area … 2 water
  - 1 use …

25

Term selection functions: document frequency

- we might now set the threshold to 2 and remove all the words that occur only once
- result: 29 words of 118 words (~25%) selected

26

Term selection: other functions

- information-theoretic term selection functions, e.g.
  - chi-square
  - information gain
  - mutual information
  - odds ratio
  - relevancy score

27

Term selection: information gain (IG)

- information gain: measures the (number of bits of) information obtained for category prediction by knowing the presence or absence of a term in a document
- information gain is calculated for each term and the best n terms with highest values are selected

28

Term selection: IG

- information gain for term t:
  - m: the number of categories

$$G(t) = -\sum_{i=1}^{m} p(c_i)\log p(c_i)$$
$$+ p(t)\sum_{i=1}^{m} p(c_i \mid t)\log p(c_i \mid t)$$
$$+ p(\sim t)\sum_{i=1}^{m} p(c_i \mid \sim t)\log p(c_i \mid \sim t)$$

29

Term selection: estimating probabilities

- Doc 1: cat cat cat (c)
- Doc 2: cat cat cat dog (c)
- Doc 3: cat dog mouse (~c)
- Doc 4: cat cat cat dog dog dog (~c)
- Doc 5: mouse (~c)

- 2 classes: c and ~c

30

Term selection: estimating probabilities

- P(t): probability of a term t
  - P(cat) = 4/5, or
    - 'cat' occurs in 4 docs of 5
  - P(cat) = 10/17
    - the proportion of the occurrences of ´cat' of the all term occurrences

31

Term selection: estimating probabilities

- P(~t): probability of the absence of t
  - P(~cat) = 1/5, or
  - P(~cat) = 7/17

32

Term selection: estimating probabilities

- $P(c_i)$: probability of category i
  - P(c) = 2/5 (the proportion of documents belonging to c in the collection), or
  - P(c) = 7/17 (7 of the 17 terms occur in the documents belonging to c)

33

Term selection: estimating probabilities

- $P(c_i \mid t)$: probability of category i if t is in the document; i.e., which proportion of the documents where t occurs belong to the category i
  - P(c | cat) = 2/4 (or 6/10)
  - P(~c | cat) = 2/4 (or 4/10)
  - P(c | mouse) = 0
  - P(~c | mouse) = 1

34

Term selection: estimating probabilities

- $P(c_i \mid \sim t)$: probability of category i if t is not in the document; i.e., which proportion of the documents where t does not occur belongs to the category i
  - P(c | ~cat) = 0 (or 1/7)
  - P(c | ~dog) = ½ (or 6/12)
  - P(c | ~mouse) = 2/3 (or 7/15)

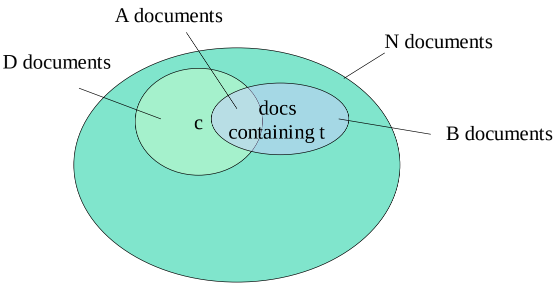
35

Term selection: estimating probabilities

- in other words...
- assume
  - term t occurs in B documents, A of them are in category c
  - category c has D documents, of the whole of N documents in the collection

36

Term selection: estimating probabilities



37

Term selection: estimating probabilities

- for instance,
  - P(t): B/N
  - P(~t): (N-B)/N
  - P(c): D/N
  - P(c|t): A/B
  - P(c|~t): (D-A)/(N-B)

38

Term selection: IG

- information gain for term t:
  - m: the number of categories

$$G(t) = -\sum_{i=1}^{m} p(c_i)\log p(c_i)$$
$$+ p(t)\sum_{i=1}^{m} p(c_i \mid t)\log p(c_i \mid t)$$
$$+ p(\sim t)\sum_{i=1}^{m} p(c_i \mid \sim t)\log p(c_i \mid \sim t)$$

39

p(c) = 2/5, p(~c) = 3/5
p(cat) = 4/5, p(~cat) = 1/5, p(dog) = 3/5, p(~dog) = 2/5,
p(mouse) = 2/5, p(~mouse) = 3/5

p(c|cat) = 2/4, p(~c|cat) = 2/4, p(c|~cat) = 0, p(~c|~cat) = 1
p(c|dog) = 1/3, p(~c|dog) = 2/3, p(c|~dog) = 1/2, p(~c|~dog) = 1/2
p(c|mouse) = 0, p(~c|mouse) = 1, p(c|~mouse) = 2/3, p(~c|~mouse) = 1/3

-(p(c) log p(c) + p(~c) log p(~c)) = -(2/5 log 2/5 + 3/5 log 3/5)
= -(2/5 (log 2 – log 5) + 3/5 (log 3 – log 5)) = -(2/5 (1 – log 5) + 3/5 (log 3 – log 5))
= -(2/5 + 3/5 log 3 – log 5) = -(0.4 + 0.96 – 2.33) = 0.97 (log base = 2)

p(cat) (p(c|cat) log p(c|cat) + p(~c|cat) log p(~c|cat))
= 4/5 (1/2 log 1/2 + 1/2 log 1/2) = 4/5 log 1/2 = 4/5 (log 1 – log 2) = 4/5 (0 – 1) = -0.8

p(~cat) (p(c|~cat) log p(c|~cat) + p(~c|~cat) log p(~c|~cat))
= 1/5 (0 + 1 log 1) = 0

G(cat) = 0.97 – 0.8 – 0 = 0.17

40

p(dog) (p(c|dog) log p(c|dog) + p(~c|dog) log p(~c|dog))
= 3/5(1/3 log 1/3 + 2/3 log 2/3) = 3/5 (1/3 (log 1 – log 3) + 2/3 (log2 - log 3))
= 3/5 (-1/3 log 3 – 2/3 log 3 + 2/3) = 3/5(-log 3 + 2/3)
= 0.6 (-1.59 + 0.67) = -0.55

p(~dog) (p(c|~dog) log p(c|~dog) + p(~c|~dog) log p(~c|~dog))
= 2/5 (1/2 log ½ + ½ log ½) = 2/5 (log 1 – log 2) = -0.4

G(dog) = 0.97 – 0.55 – 0.4 = 0.02

p(mouse) (p(c|mouse) log p(c|mouse) + p(~c|mouse) log p(~c|mouse))
= 2/5 (0 + 1 log 1) = 0

p(~mouse) (p(c|~mouse) log p(c|~mouse) + p(~c|~mouse) log p(~c|~mouse))
= 3/5 (2/3 log 2/3 + 1/3 log 1/3) = -0.55

G(mouse) = 0.97 – 0 – 0.55 = 0.42

ranking: 1. mouse 2. cat 3. dog

41

Example, some intuitive remarks

- 'mouse' is the best, since it occurs in ~c documents only
- 'cat' is good, since if it does not occur, the category is always ~c
- 'cat' is not good, since half of the documents in which 'cat' occurs are in c, half are in ~c
- 'dog' is the worst, since if it occurs, the category can be either c or ~c, and if it does not occur, the category can also be either c or ~c

42