Processing of large document collections

Part 4 (Applications of text categorization, boosting, text summarization)

Helena Ahonen-Myka

Spring 2006

In this part

- Applications of text categorization
- Classifier committees, boosting
- Text summarization

Applications of text categorization

- automatic indexing for Boolean information retrieval systems
- document organization
- text filtering
- word sense disambiguation
- authorship attribution
- hierarchical categorization of Web pages

Automatic indexing for information retrieval systems

- in an information retrieval system, each document is assigned one or more keywords or keyphrases describing its content
  - keywords may belong to a finite set called controlled dictionary
- text categorization problem: the entries in a controlled dictionary are viewed as categories
  - $k_1 \leq x \leq k_2$ keywords are assigned to each document

Document organization

- indexing with a controlled vocabulary is an instance of the general problem of document collection organization
- e.g. a newspaper office has to classify the incoming ”classified” ads under categories such as Personals, Cars for Sale, Real Estate etc.
- organization of patents, filing of newspaper articles...

Text filtering

- classifying a stream of incoming documents by an information producer to an information consumer
- e.g. newsfeed
  - producer: news agency; consumer: newspaper
  - the filtering system should block the delivery of documents the consumer is likely not interested in

Word sense disambiguation

- given the occurrence in a text of an ambiguous word, find the sense of this particular word occurrence
- e.g.
  - bank, sense 1, like in ”Bank of Finland”
  - bank, sense 2, like in ”the bank of river Thames”
  - occurrence: ”Last week I borrowed some money from the bank.”

7

Word sense disambiguation

- indexing by word senses rather than by words
- text categorization
  - documents: word occurrence contexts
  - categories: word senses
- also resolving other natural language ambiguities
  - context-sensitive spelling correction, part of speech tagging, prepositional phrase attachment, word choice selection in machine translation

8

Authorship attribution

- task: given a text, determine its author
- author of a text may be unknown or disputed, but some possible candidates and samples of their works exist
- literary and forensic applications
  - who wrote this sonnet? (literary interest)
  - who sent this anonymous letter? (forensics)

9

Hierarchical categorization of Web pages

- e.g. Yahoo like web hierarchical catalogues
- typically, each category should be populated by ”a few” documents
- new categories are added, obsolete ones removed
- usage of link structure in classification
- usage of the hierarchical structure

10

More learning methods: classifier committees

- idea: given a task that requires expert knowledge, S independent experts may be better than one if their individual judgments are appropriately combined
- idea can be applied to text categorization
  - apply S different classifiers to the same task of deciding under which set of categories a document should be classified

11

Classifier committees

- usually, the classifiers are different
  - either in terms of text representation (indexing, term selection)
  - or in terms of a learning method
  - or both
- a classifier committee is characterized by
  - a choice of S classifiers
  - a choice of a combination function

12

Boosting

- the boosting method uses a committee of classifiers, but
  - the classifiers are obtained by the same learning method
  - the classifiers are not parallel and indepent, but work sequentially
    - a classifier may take into account how the previous classifiers perform on the training documents
    - and concentrate on getting right those training documents on which the previous classifiers performed worst
  - the classifiers work on the same text representation

13

Boosting

- the main idea of boosting:
  - combine many weak classifiers to produce a single highly effective classifier
- example of a weak classifier: "if the word 'money' appears in the document, then predict that the document belongs to category c"
  - this classifier will probably misclassify many documents, but a combination of many such classifiers can be very effective
- one boosting algorithm: AdaBoost

14

AdaBoost

- assume: a training set of pre-classified documents (as before)
- boosting algorithm calls a weak learner T times (T is a parameter)
  - each time the weak learner returns a classifier
  - error of the classifier is calculated using the training set
  - weights of training documents are adjusted
    - "hard" examples get more weight
  - the weak learner is called again
- finally the weak classifiers are combined

15

AdaBoost: algorithm

- Input:
  - N documents and labels: $<(d_1,y_1), \ldots,(d_N, y_N)>$, where $y_i \in \{-1, +1\}$ (-1=false, +1=true)
  - integer T: the number of iterations
- Initialize $D_1(i)$: $D_1(i) = 1/N$
- For s = 1,2,…,T do
  - Call WeakLearn and get a weak hypothesis h_s
  - Calculate the error of h_s: ε_s
  - Update the distribution (weights) of examples: $D_s(i)$ -> $D_{s+1}(i)$
- Output the final hypothesis

16

Distribution of examples

- Initialize $D_1(i)$: $D_1(i) = 1/N$
- if N = 10 (there are 10 documents in the training set), the initial distribution of examples is:
  - $D_1(1) = 1/10$, $D_1(2) = 1/10$, …, $D_1(10) = 1/10$
- the distiribution describes the importance (=weight) of each example
- in the beginning all examples are equally important
  - later "hard" examples are given more weight

17

WeakLearn

- idea: a classifier consists of one rule that tests the occurrence of one term
  - document d is in category c if and only if d contains this term
- to find the best term, the weak learner computes for each term the error

$$\varepsilon(t) = \sum_{i:t\in d_i, d_i\notin c} D_s(i) + \sum_{i:t\notin d_i, d_i\in c} D_s(i)$$

  - a good term discriminates between positive and negative examples
    - both occurrence and non-occurrence of a term can be significant

18

WeakLearn

- a term is chosen that minimizes ε(t) or 1- ε(t)
- let t_s be the chosen term
- the classifier h_s for a document d:

$$h_s(d) = \begin{cases} +1 & \text{if } t_s \in d \\ -1 & \text{if } t_s \notin d \end{cases}$$

19

Calculate the error

- calculate the error of h_s

$$\varepsilon_s = \sum_{i:h_s(d_i) \neq y_i} D_s(i)$$

- error = the sum of the weights of false positives and false negatives (in the training set)

20

Update weights

- the weights of training documents are updated
  - documents classified correctly get a lower weight
  - misclassified documents get a higher weight

$$D_{s+1}(i) = \frac{D_s(i)}{Z_s} \times \begin{cases} e^{-\alpha_s} & \text{if } h_s(d_i) = y_i \\ e^{\alpha_s} & \text{if } h_s(d_i) \neq y_i \end{cases}$$

21

Update weights

- calculation of α_s:

$$\alpha_s = \frac{1}{2} \ln\left(\frac{1-\varepsilon_s}{\varepsilon_s}\right)$$

- if error is small (<0.5), α_s is positive
- if error is 0.5, α_s=0
- if error is large (>0.5), α_s is negative

22

Update weights

- if error is small, then α_s is large
  - if d_i correctly classified, then the weight is decreased drastically
  - if d_i is not correctly classified, then the weight is increased drastically
- if error is 0.5, then α_s =0
  - weights do not change
- if error is close to 0.5 (e.g. 0.4), then α_s is small but positive
  - if d_i correctly classified, then the weight is decreased slightly (multiplied by 0.82)
  - if d_i is not correctly classified, then the weight is increased slightly (multiplied by 1.22)

23

Update weights

- Z_s is a normalization factor
  - the weights have to form a distribution also after updates -> the sum of weights has to be 1

24

Final classifier

- the decisions of all weak classifiers are evaluated on the new document d and combined by voting:

$$h_{fin}(d) = \begin{cases} +1 & \text{if } \sum_{s=1}^{T} \alpha_s h_s(d) > 0 \\ -1 & \text{otherwise} \end{cases}$$

- note: α_s is also used to represent the goodness of the classifier s

25

Performance of AdaBoost

- Schapire, Singer and Singhal (1998) have compared AdaBoost to Rocchio's method in text filtering
- experimental results:
  - AdaBoost is more effective, if a large number (hundreds) of documents are available for training
    - otherwise no noticeable difference
  - Rocchio is significantly faster

26

4. Text summarization

- "Process of distilling the most important information from a source to produce an abridged version for a particular user or task" (Mani, Maybury, 1999)

27

Text summarization

- many everyday uses:
  - news headlines
  - minutes (of a meeting)
  - tv digests
  - reviews (of books, movies)
  - abstracts of scientific articles
  - ...

28

American National Standard for Writing Abstracts (1)

[Cremmins 82, 96]

- State the purpose, methods, results, and conclusions presented in the original document, either in that order or with an initial emphasis on results and conclusions.
- Make the abstract as informative as the nature of the document will permit, so that readers may decide, quickly and accurately, whether they need to read the entire document.
- Avoid including background information or citing the work of others in the abstract, unless the study is a replication or evaluation of their work.

29

American National Standard for Writing Abstracts (2)

[Cremmins 82, 96]

- Do not include information in the abstract that is not contained in the textual material being abstracted.
- Verify that all quantitative and qualitative information used in the abstract agrees with the information contained in the full text of the document.
- Use standard English and precise technical terms, and follow conventional grammar and punctuation rules.
- Give expanded versions of lesser known abbreviations and acronyms, and verbalize symbols that may be unfamiliar to readers of the abstract
- Omit needless words, phrases, and sentences.

30

Example

- Original version:

  There were significant positive associations between the concentrations of the substance administered and mortality in rats and mice of both sexes.

  There was no convincing evidence to indicate that endrin ingestion induced and of the different types of tumors which were found in the treated animals.

- Edited version:

  Mortality in rats and mice of both sexes was dose related.

  No treatment-related tumors were found in any of the animals.

31

Input for summarization

- a single document or multiple documents
- text, images, audio, video
- database

32

Characteristics of summaries

- extract or abstract
  - extract: created by reusing portions (usually sentences) of the input text verbatim
  - abstract: may reformulate the extracted content in new terms
- compression rate
  - ratio of summary length to source length
- connected text or fragmentary
  - extracts are often fragmentary

33

Characteristics of summaries

- generic or user-focused/domain-specific
  - generic summaries:

    summaries addressing a broad, unspecific user audience, without considering any usage requirements
  - tailored summaries:

    summaries addressing group specific interests or even individualized usage requirements or content profiles
    - expressed via query terms, interest profiles, feedback info, time window

34

Characteristics of summaries

- query-driven or text-driven summary
  - top-down: query-driven focus
    - criteria of interest encoded as search specifications
    - system uses specifications to filter or analyze relevant text portions.
  - bottom-up: text-driven focus
    - generic importance metrics encoded as strategies.
    - system applies strategies over representation of whole text.

Characteristics of summaries

- indicative, informative, or critical summaries
  - indicative summaries

    summary has a reference function for selecting relevant documents for in-depth reading
  - informative summaries

    summary contains all the relevant (novel) information of the original document, thus substituting the original document
  - critical summaries

    summary not only contains all the relevant information but also includes opinions, critically assesses the quality of and the major assertions expressed in the original document

36

Architecture of a text summarization system

- three phases:
  - analyzing the input text
  - transforming it into a summary representation
  - synthesizing an appropriate output form

37

The level of processing

- surface level
- discourse level

38

Surface-level approaches

- tend to represent text fragments (e.g. sentences) in terms of shallow features
- the features are then selectively combined together to yield a salience function used to select some of the fragments

39

Surface level

- Shallow features of a text fragment
  - thematic features
    - presence of statistically salient terms, based on term frequency statistics
  - location
    - position in text, position in paragraph, section depth, particular sections
  - background
    - presence of terms from the title or headings in the text, or from the user's query

40

Surface level

- Cue words and phrases
  - "in summary", "our investigation"
  - emphasizers like "important", "in particular"
  - domain-specific bonus (+) and stigma (-) terms

41

Discourse-level approaches

- model the global structure of the text and its relation to communicative goals
- structure can include:
  - format of the document (e.g. hypertext markup)
  - threads of topics as they are revealed in the text
  - rhetorical structure of the text, such as argumentation or narrative structure

42