Eleventh Conference of the European Chapter
of the Association for Computational Linguisitics
EACL 2003 - http://www.conferences.hu/EACL03/

Text Representation for Automatic Text Categorization

April 12, 2003

José María Gómez Hidalgo
Departamento de Inteligencia Artificial
Artificial Intelligence Department
Universidad Europea de Madrid
jmgomez@dinar.esi.uem.es
http://www.esi.uem.es/~jmgomez/

This tutorial is supported by:
European Commission, under the Safer Internet Action Plan project POESIA (2117)
See http://www.poesia-filter.org

Web page
http://www.esi.uem.es/~jmgomez/tutorials/eacl03/index.html

Text Representation for Automatic Text Categorization

José María Gómez Hidalgo
Departamento de Inteligencia Artificial
Universidad Europea de Madrid
`jmgomez@dinar.esi.uem.es`
`http://www.esi.uem.es/~jmgomez/`

This tutorial is partly funded by the European Commission, under the Safer Internet Action Plan project POESIA (2117)
See `http://www.poesia-filter.org`

Outline

1. Automated Text Categorization (ATC)
2. Applications
3. A blueprint for learning-based ATC
4. Advanced document indexing
5. Task oriented features
6. Summary

1. Automated Text Categorization

1. Automated Text Categorization

- Text Categorization (TC) = assignment of documents to *predefined* classes
- *Documents* can be news stories, technical reports, web pages, e-mail messages, books
- *Categories* are most often subjects or topics (e.g. ARTS, ECONOMY), but may be based on style (genres), pertinence (spam e-mail, adult web pages), etc

1. Automated Text Categorization

- It is *a Text Mining subtask* [Hearst99], as Information Retrieval or Filtering, Document Clustering, etc.
- Taxonomy of Text Mining subtasks based on [Lewis92], according to several dimensions
  - Size of text
  - Involve supervised or unsupervised learning
  - Text classification vs. understanding
    - Assigning documents or parts to a number of groups vs.
    - More complex access to document content
    - Note it is not a sharp division

1. Automated Text Categorization

- Sample text classification tasks

	Words	Documents
Supervised learning	POS Tagging, Word Sense Disambiguation	Text Categorization, Filtering, Topic Detection and Tracking
Unsupervised learning	Latent Semantic Indexing, Automatic Thesaurus Construction, Key Phrase Extraction	Document Clustering, Topic Detection and Tracking

1. Automated Text Categorization

- Sample text understanding tasks

	Words	Documents
Supervised learning		Information Extraction
Unsupervised learning	Word Sense Discovery	Summarization

1. Automated Text Categorization

- TC is often manual, requiring skilled specialists
  - Library cataloguers (e.g. National Library of Medicine has more than 200)
  - Web directory editors (e.g. dmoz.org (>3000), Yahoo! (>100))
- The goal is to (semi) automate it for
  - Reducing cost
  - Improving performance (including accuracy and consistency)

1. Automated Text Categorization

- The two main trends for automation are
  - *Knowledge based* approach
    - Knowledge about classification is obtained from experts and codified in the form of classification rules
  - *Learning based* approach
    - Experts are requested not to explain but to classify examples
    - Information Retrieval (IR) and Machine Learning (ML) techniques used to induce an automatic classifier
    - The knowledge acquisition problem is reduced

1. Automated Text Categorization

- The problem can be defined as
  - Given a set of documents D and a set of categories C
  - To approximate an unknown classification function Φ:DxC$\rightarrow$Boolean defined as

$$\Phi(d,c) = \begin{cases} \mathit{true} & \text{if } d \in c \\ \mathit{false} & \text{otherwise} \end{cases}$$

  - For any pair (*d,c*) of document and category

1. Automated Text Categorization

$$
\begin{aligned}
\overline{\Phi}(d, \text{WHEAT}) = {} & (\text{"wheat"} \in d \wedge \text{"farm"} \in d) \vee \\
& (\text{"wheat"} \in d \wedge \text{"commodity"} \in d) \vee \\
& (\text{"bushels"} \in d \wedge \text{"export"} \in d) \vee \\
& (\text{"wheat"} \in d \wedge \text{"tonnes"} \in d) \vee \\
& (\text{"wheat"} \in d \wedge \text{"winter"} \in d \wedge \text{"soft"} \notin d)
\end{aligned}
$$

Sample of rule [Apte94], similar to those used in the Construe system, developed by Carnegie Group for Reuters [Hayes90], for category WHEAT

1. Automated Text Categorization

- Kinds of categories
  - Organization
    - Hierarchical (e.g. Yahoo!, MEdical Subject Headings - MESH, personal e-mail folders)
    - Flat (e.g. newspaper sections, Reuters-21578 topics)
  - Membership of documents (a documents belongs to exactly one or to several categories)
    - Overlapping (e.g. Reuters-21578 topics, MESH)
    - Disjoint (e.g. personal e-mail folders, newspaper sections)

2. Applications

2. Applications

- Information/knowledge access/management
  - Maintaining a directory of documents
    - Helps to provide an uniform communication vocabulary (e.g. for intranet/Internet portals [Adams01, Chen00, Labrou99])
    - Helps to search by providing context to results (e.g. the category links provided by Google) [Hearst94]
      - Yahoo! demo [Mladenic98]
      - SWISH [Chen00]

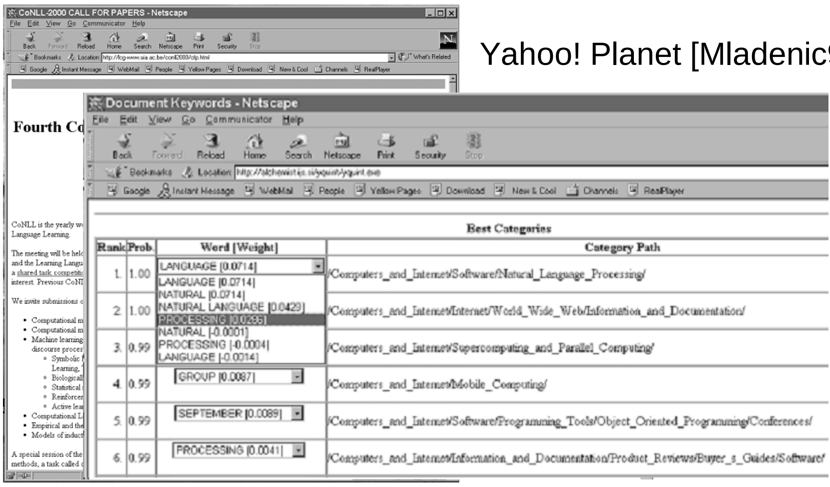
2. Applications

- Information/knowledge access/management
  - (Semi)automatic library cataloging (e.g. patent filing in [Larkey99], med records in [Larkey96])
  - Information Filtering
    - Recommendation
      - Setting filter profile in terms of categories (e.g. News Stories in Mercurio & Hermes [Diaz01, Giraldez02])
    - Blocking
      - Blocking spam e-mail (e.g. [Gomez02]) and adult Web content (e.g. POESIA [Gomez02c])

2. Applications

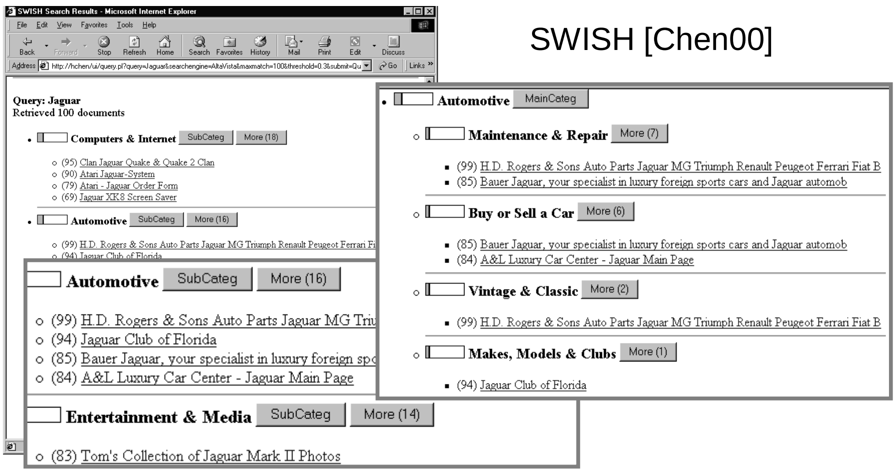
- Information/knowledge access/management
  - Personal information management
    - Organizing files (e.g. SONIA [Sahami98])
    - Organizing e-mail messages in folders (e.g. SwiftFile [Segal99, Segal00])
- Language [Cavnar94], genre [Kessler97, Stamatatos00] and authorship [Forsyth99, Teahan00] identification
- Automatic essay grading [Larkey98]
- See [Sebastiani02] for more

2. Applications



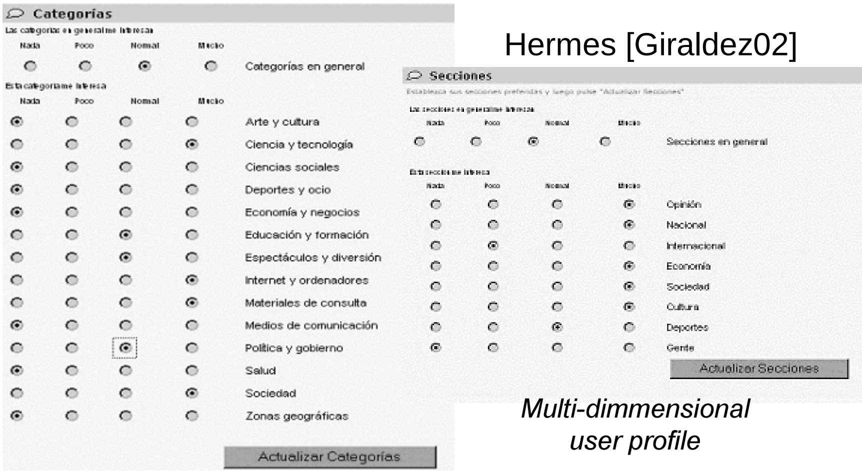
Yahoo! Planet [Mladenic98]

2. Applications



SWISH [Chen00]

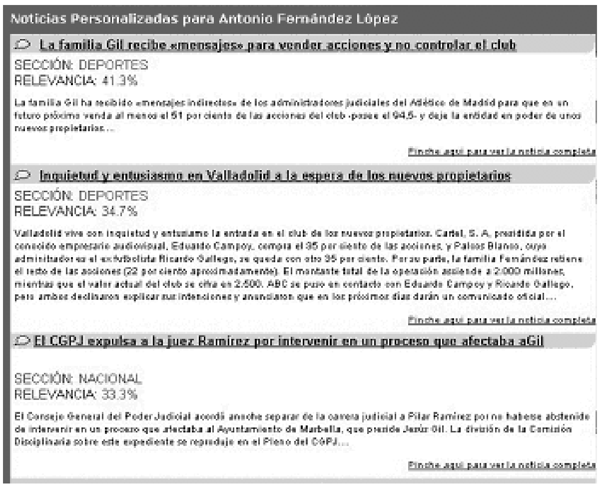
2. Applications



Hermes [Giraldez02]

Multi-dimmensional user profile

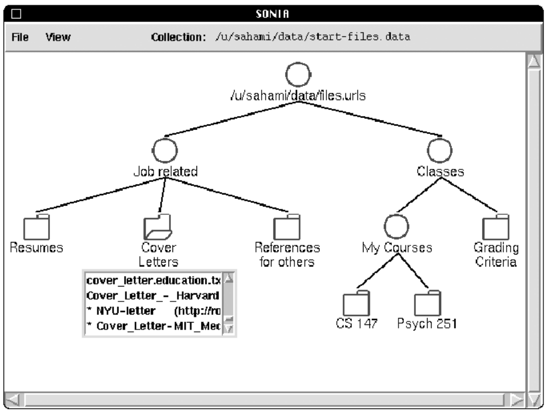
2. Applications



Hermes [Giraldez02]

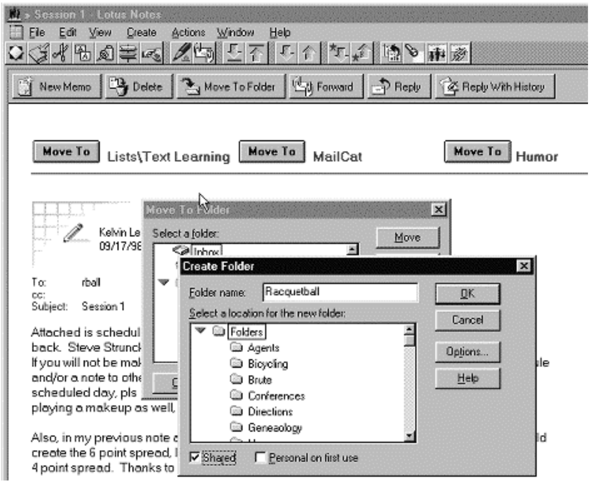
2. Applications

2. Applications



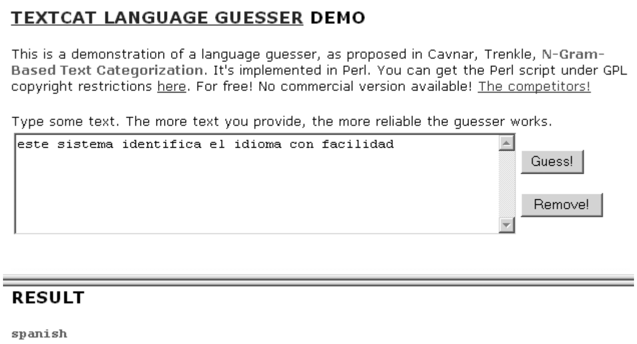
SONIA
[Sahami98]

2. Applications



SwiftFile
[Segal99,00]

2. Applications



TextCat
(based on
[Cavnar94])

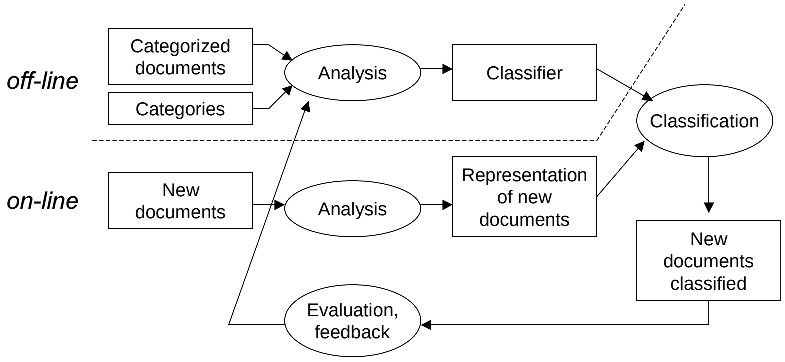
3. A blueprint for learning-based ATC

3. A blueprint for learning-based ATC

- A simple model for learning based ATC (following Salton's blueprint for automatic indexing [Salton89])
  - As effective as manual thematic TC
  - Based on IR & ML techniques
  - Requires a set of manually classified documents (training collection)
    - Depends on the number and quality of training documents
    - Assumes that new documents will be training-like

3. A blueprint for learning-based ATC

- Process (Belkin's way [Belkin92])



3. A blueprint for learning-based ATC

- Evaluation must be addressed first!
- As in IR, most evaluation issues in NLP systems (e.g. [SparckJones95]) are ignored
- ATC researchers focus on
  - Effectiveness
    - Addresses the quality of the approximation to the unknown Φ function
  - Efficiency
    - Theoretical and practical time and memory requirements for learning and classification

3. A blueprint for learning-based ATC

- Effectiveness
  - Some available (manually classified) benckmark collections include
    - Reuters-21578
    - The Reuters Corpus Volume 1
    - OHSUMED
    - 20-NewsGroups
    - Ling-Spam
  - The collection is split into two parts, one for training and one for testing
  - Cross-validation is not frequent

3. A blueprint for learning-based ATC

- Effectiveness
  - Standard IR & ML metrics

	Actual	
System	C	¬C
C	tp	fp
¬C	fn	tn

Contingency matrix

$$\text{recall}\,(r) = \frac{tp}{tp+fn} \qquad \text{precision}\,(p) = \frac{tp}{tp+fp}$$

$$\text{accuracy} = \frac{tp+tn}{tp+fn+fp+tn}$$

$$F_\beta = \frac{1}{\beta\frac{1}{p}+(1-\beta)\frac{1}{r}} \qquad F_1 = \frac{2pr}{p+r}$$

3. A blueprint for learning-based ATC

- Effectiveness
  - In multi class situations, at least report F_1 by
    - *Macro averaging* (*M*) – averaging on the number of classes
      - All categories are equally important
    - *Micro averaging* (*m*) – computing over all decisions at once
      - More populated categories are more important
  - Scarce statistical testing (intro in [Yang99])
  - *Accuracy* and *error* do not fit well TC because class distribution is usually highly biased
  - Now increasing use of cost-sensitive metrics for specific tasks (e.g. cost, ROCCH method [Gomez02])

3. A blueprint for learning-based ATC

- Effectiveness (an example)
  - Given categories C1, C2, and 100 test docs

Sys	C1 (Actual)	¬C1 (Actual)
C1	30	5
¬C1	10	55

$r(C1) = .75$
$p(C1) = .85$
$F_1(C1) = .80$

Sys	C2 (Actual)	¬C2 (Actual)
C2	2	10
¬C2	3	85

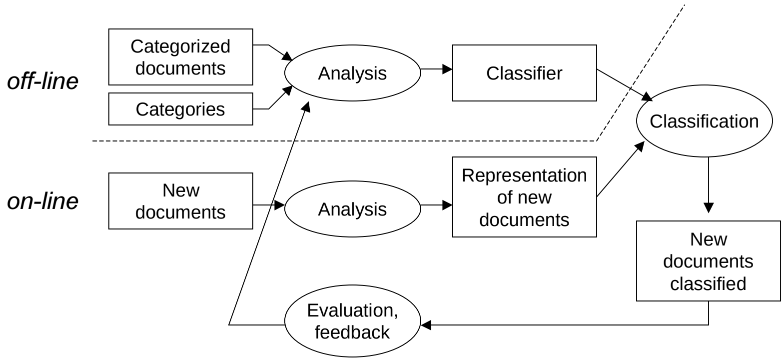
$r(C2) = .60$
$p(C2) = .20$
$F_1(C2) = .30$

Sys	C (Actual)	¬C (Actual)
C	32	15
¬C	13	140

$F_1^m = .69$
$F_1^M = .55$

3. A blueprint for learning-based ATC

- The process again



3. A blueprint for learning-based ATC

1. Analysis of training documents
   1. Building a representation (*indexing*)
      1. Obtaining a set of representing concepts (terms, words...) – features, and weights – values
      2. Reducing the dimensionality (term selection & extraction)
   2. Learning a classifier
2. Analysis of new documents according to the training documents representation
3. Classifying new documents with the learned classifier

3. A blueprint for learning-based ATC

1. Basic representation
   - Often named *bag-of-words*
   - Corresponds to Salton's Vector Space Model (VSM) [Salton89]
   - Each document is represented as a term-weight vector in which
     - Terms or concepts are usually (stemmed, stoplist filtered) words
     - Weights are binary (0 or 1), TF (term-frequency) or TF.IDF (term-frequency, inverse document frequency)

3. A blueprint for learning-based ATC

1. Basic representation
   - Basic concepts are words (minimal meaningful units)
   - IR Stoplist filtering aims at eliminating low content words (adverbs, prepositions, etc.)
   - IR Stemming (e.g. [Porter80]) aims at obtaining canonical word forms (analyzing, analyzer, analysis => analy)
   - Side effect => reducing vocabulary size

3. A blueprint for learning-based ATC

1. Basic representation
   – Stoplist filtering and stemming may hurt categorization accuracy
   – E.g. [Riloff95]

     1200 news stories dealing or not with JOINT VENTURES

Words	Recall	Precision
joint, venture	93.3%	88.9%
tie-up	2.5%	84.2%
venture	95.5%	82.8%
jointly	11.0%	78.9%
joint-venture	6.4%	73.2%
consortium	3.6%	69.7%
joint, ventures	19.3%	66.7%
partnership	7.0%	64.3%
ventures	19.8%	58.8%

3. A blueprint for learning-based ATC

1. Basic representation
   – Given a set D of documents and a set T of terms, the weight wd_{ij} of term t_i in document d_j can be

binary $$wd_{ij} = \begin{cases} 1 & \text{if } t_i \text{ occurs in } d_j \\ 0 & \text{otherwise} \end{cases}$$

TF $$wd_{ij} = tf_{ij}$$

TF.IDF $$wd_{ij} = tf_{ij} \cdot \log_2\left(\frac{|D|}{df_i}\right)$$

Being
tf_{ij} the # of times that t_i occurs in d_j
df_i the # of documents in which t_i occurs

3. A blueprint for learning-based ATC

1. Basic representation
   - Assuming that

     Stoplist = {are, and, be, by, or}

     Stemmed concept set T = {available, currenc, dollar, earn, pound}

     |D| = 200

     df_1 = 100, df_2 = 200, df_3 = 50, df_4 = 100, df_5 = 25

     (=> idf_1 = 1, idf_2 = 0, idf_3 = 2, idf_4 = 1, idf_5 = 3)

3. A blueprint for learning-based ATC

1. Basic representation
   - The document "Available currencies are US dollars, UK pounds and HK dollars" is represented as

$$\begin{aligned}\vec{d}_{bin} &= \langle 1,1,1,0,1 \rangle \\ \vec{d}_{TF} &= \langle 1,1,2,0,1 \rangle \\ \vec{d}_{TF.IDF} &= \langle 1,0,4,0,3 \rangle\end{aligned}$$

3. A blueprint for learning-based ATC

2. Dimensionality Reduction (DR)
   - The goal is to reduce the number of concepts to
     - Keep or increase effectiveness
     - Reduce learning time
     - Avoid over fitting
   - Not all learning methods require it (e.g. Support Vector Machines)
   - It can be
     - Feature selection – a subset of the original set
     - Feature extraction – a set of new features

3. A blueprint for learning-based ATC

2. (DR) Feature (concept, term) Selection
   - Keep best features according to a quality metric
   - The metric should score high the most informative-predictive-separating concepts
   - Given a category *C*, a "perfect" concept should occur in a document *d* if and only if $d \in C$, or if and only if $d \notin C$
     - e.g. Most spam messages claim "this is not spam", and none of personal messages do
     - e.g. delete low frequency terms

3. A blueprint for learning-based ATC

2. (DR) Feature Selection
   - Some effective quality metrics include
     - Information Gain - IG

       $$IG(c,t) = \sum_{x \in \{c,\bar{c}\}} \sum_{y \in \{t,\bar{t}\}} P(x,y) \cdot \log_2 \frac{P(x,y)}{P(x) \cdot P(y)}$$

       Being *t* a concept and *c* a category
     - Document Frequency – DF, the number of documents in which the concept occurs
       - Highly related to IR's discrimination power

3. A blueprint for learning-based ATC

2. (DR) Feature Selection
   - Several more including odds ratio, χ^2 [Sebastiani02], with variable effectiveness
   - For instance, from [Yang97]
     - IG and χ^2 are very effective (allow to eliminate 99% of concepts without effectiveness decrease in classification)
     - DF is quite effective (90% elimination)
     - Mutual Information and Term Strength are bad

3. A blueprint for learning-based ATC

2. (DR) Feature Selection
   - class dependent metrics can be averaged over all classes
   - Given a metric denoted by Q(t,c), being t a concept and c a class in a set C, several possible averages including

$$Q_{avg}(t) = \sum_{c \in C} P(c) Q(t,c)$$

$$Q_{\max}(t) = \max_{c \in C}\{Q(t,c)\}$$

3. A blueprint for learning-based ATC

2. (DR) Feature Extraction
   - Concept clustering as usual in IR (e.g. [Salton89]) = automatic thesaurus construction
     - Class #764 of an engineering related thesaurus
       (refusal) refusal declining non-compliance rejection denial
   - Latent semantic indexing (e.g. [Dumais92])
     - a way to capture main semantic dimensions in a text collection, avoiding *synonymy* and *polysemy* problems
     - Mapping a high-dimensional space into a low-dimensional one, iteratively choosing dimensions corresponding to the axes of greater variation

3. A blueprint for learning-based ATC

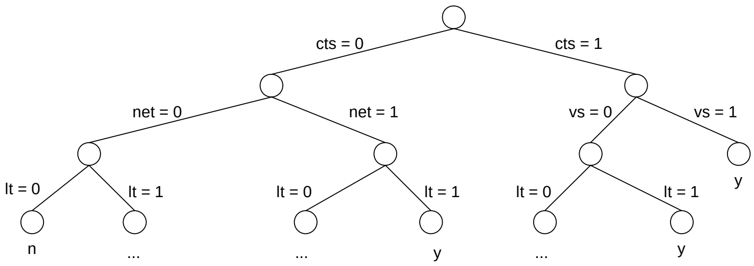
3. Learning TC classifiers
   - In order to approximate Φ, many learning algorithms have been applied, including
     - Probabilistic classifiers as Naive Bayes [Lewis92]
     - Decision tree learners as C4.5 [Cohen98]
     - Rule learners as Ripper [Cohen95]
     - Instance-based classifiers as kNN [Larkey98]
     - Neural networks [Dagan97]
     - Support Vector Machines (SVM) [Joachims98]
     - etc.

3. A blueprint for learning-based ATC

3. Learning TC classifiers (example)
   – Category EARN (*earnings*) in the Reuters-21578 benchmark collection (ModApte split)
   – 9,606 training documents (2,879 in EARN)
   – 3,299 test documents (1,087 in EARN)
   – Documents represented as binary vectors
   – Selected top five χ^2 scoring terms ("cts", "net", "lt", "loss", "vs")

3. A blueprint for learning-based ATC

3. Learning TC classifiers (example)



A (part of a) decision tree generated by ID3 using the WEKA [Witten99] package (*the tree captures context information*)

3. A blueprint for learning-based ATC

3. Learning TC classifiers (example)

$$(\text{"cts"} \in D) \wedge (\text{"vs"} \in D) \rightarrow EARN$$
$$(\text{"net"} \notin D) \wedge (\text{"cts"} \notin D) \wedge (\text{"loss"} \notin D) \rightarrow \overline{EARN}$$
$$(\text{"lt"} \notin D) \wedge (\text{"vs"} \notin D) \wedge (\text{"cts"} \notin D) \rightarrow \overline{EARN}$$
$$(\text{"lt"} \in D) \rightarrow EARN$$
$$(\text{"net"} \notin D) \rightarrow \overline{EARN}$$
$$(\text{"loss"} \notin D) \rightarrow \overline{EARN}$$
$$T \rightarrow EARN$$

A list of rules produced by PART using WEKA (*rules capture context information*)

3. A blueprint for learning-based ATC

3. Learning TC classifiers (example)

$$f_{EARN}(d) = -2{,}000 \cdot wd_1 - 1{,}998 \cdot wd_2 - 0{,}001 \cdot wd_3 + \\ -0{,}001 \cdot wd_4 - 0{,}002 \cdot wd_5 + 1{,}002$$

A linear function generated by SVM using WEKA

Assign a document d to EARN if and only if $f_{EARN}(d) \geq 0$

(which means EARN is the default case unless "cts" or "net" occur in the document)

(*the linear function does not capture context inform.*)

3. A blueprint for learning-based ATC

3. Learning TC classifiers (example)

Algorithm	**Pr**	**Re**	**F1**	**Acc**
NaiveBayes	0,916	0,927	0,921	0,947
ID3	*0,913*	*0,938*	*0,926*	*0,950*
PART	*0,914*	*0,938*	*0,926*	*0,950*
1NN	0,613	0,926	0,737	0,782
2NN	0,913	0,938	0,926	0,950
5NN	*0,914*	*0,938*	*0,926*	*0,950*
SVM	0,866	0,936	0,899	0,930

4. Advanced document indexing

4. Advanced document indexing

1. Introduction
2. Statistical and linguistic phrases
3. Information Extraction patterns
4. Using WordNet

4. Advanced document indexing
4.1. Introduction

- A number of approaches aim at enriching text representation for general purpose ATC
  - To better capture text semantics
- They can be seen as feature extraction
- Typically, mixed results in experiments
- We will not cover
  - Using unlabelled documents for improving word statistics (e.g. [McCallum98, Zelikovitz01])

4. Advanced document indexing
4.2. Statistical and linguistic phrases

- Many works have proposed the use of phrases as indexing concepts
- Phrases = good indexing concepts in IR when
  - Text collections are specialized (e.g. Medicine, computer science)
  - Individual terms are too frequent [Salton89]
- Phrases can be
  - Statistical – Normalized n-grams
  - Linguistic – Noun phrases

4. Advanced document indexing
4.2. Statistical and linguistic phrases

- Statistical phrases [Caropreso01]
  - Defined as n-grams normalized with stoplist filtering, stemming and alphabetical ordering, e.g.

$$\left.\begin{array}{r}\text{“information retrieval”}\\ \text{“retrieval of information”}\\ \text{“retrieved information”}\\ \text{“informative retrieval”}\end{array}\right\}\Rightarrow \text{“inform retriev”}$$

  - May show
    - Over-generalization – No valid concepts
    - Under-generalization – Valid concepts missed

4. Advanced document indexing
4.2. Statistical and linguistic phrases

- Statistical phrases [Caropreso01]
  - Classifier independent evaluation as penetration of 2-grams
    - Percentage of selected concepts that are 2-grams, by using several selection metrics (IG, χ^2, DF, etc.) per category or averaged
  - It is shown that
    - Penetration levels are high – 2-grams valuable
    - Increasing reduction decreases penetration

4. Advanced document indexing
4.2. Statistical and linguistic phrases

- Statistical phrases [Caropreso01]
  - Direct evaluation with the Rocchio algorithm
  - Results are
    - In 20 of 48 cases, adding 2-grams hurts performance
    - Most improvements are got at bigger concept number
  - Some 2-grams may be redundant, and force the elimination of valuable 1-grams
    - E.g. “inform”, “retriev” and “inform retriev” are all selected

4. Advanced document indexing
4.2. Statistical and linguistic phrases

- Statistical phrases
  - More work for ATC in e.g. [Furnkranz98, Lewis92, Mladenic98b, Mladenic98c, Scott98, Scott99]
  - Work in IR is also relevant (specially from [Fagan87, Fagan89] ahead)
  - Mixed results, maybe because indexing languages based on phrases have, with respect to word-only indexing languages
    - superior semantic qualities
    - inferior statistical qualities

4. Advanced document indexing
4.2. Statistical and linguistic phrases

- Linguistic phrases
  - Concepts *often* include Noun Phrases, recognized by statistical methods, involving POS-Tagging and
    - Chunking (shallow parsing)
      - E.g. In [Lewis92, Lewis92b], the *parts* bracketer [Church88] is used
    - Finite state methods (e.g. regular expressions)
      - E.g. [Scott99]

        NP = {A, N}* N

4. Advanced document indexing
4.2. Statistical and linguistic phrases

- Linguistic phrases
  - In [Lewis92], syntactic phrases do not outperform terms as indexing concepts, for a Naive Bayes classifier for Reuters-21578
  - In [Scott99], there is a slight improvement for the rule learner Ripper [Cohen95] for Reuters-21578
  - Remarks on statistical phrases hold also here

4. Advanced document indexing
4.3. Information extraction patterns

- Riloff's relevancy signatures [Riloff94,Riloff96]
  - Signatures are <word, semantic_node> pairs
  - Words act as semantic node triggers
  - Semantic nodes are manually defined for a domain
  - Patterns are detected with the CIRCUS sentence analyzer
    e.g. Terrorism incidents (MUC)

Signature	**P(c|s)**	**Example**
<*assassination*, $murder$>	.84	the assassination of Hector Oqueli
<*assassinations*, $murder$>	.49	there were 2,978 political assassinations in 1988
<*dead*, $found-dead-pasive$>	1.00	the major was found dead
<*dead*, $left-dead$>	.61	the attack left 9 people dead

4. Advanced document indexing
4.3. Information extraction patterns

- Riloff's relevancy signatures [Riloff94,Riloff96]
  - The relevancy signatures operates as follows
  - *Training* (being *C* a category, *S* a signature)
    - Collect all signatures from training texts
    - Select those with *P(C|S)* > *R*, and occurring more than *M* times => a set *S* of "relevancy signatures"
  - Signatures in S have reliable statistics (*M*) and guarantee high precision (*R*)
  - *Classification*
    - Collect signatures from the document *D* to classify
    - Classify it in *C* if and only if a relevancy signature occurs in *D*

4. Advanced document indexing
4.3. Information extraction patterns

- Riloff's relevancy signatures [Riloff94,Riloff96]
  - Evaluation results on several kinds of problems
    - Detecting terrorist attacks
    - Detecting joint venture events
    - Finding microelectronic processes linked to specific organizations
  - Results consistently show high precision for low recall levels
  - The main drawback is manually writing semantic nodes (a knowledge acquisition bottleneck) alleviated with semiautomatic programs (AutoSlog)

4. Advanced document indexing
4.3. Information extraction patterns

- [Furnkranz98]
  - The AutoSlog-TS [Riloff96] IE system is used for extracting phrases matching syntactic patterns

Syntactic pattern	**Phrasal feature**
noun aux-verb <d-obj>	I am <_>
<subj> aux-verb noun	<_> is student
noun verb <noun-phrase>	student of <_> student at <_>

In "I am a student of computer science at Carnegie Mellon University", 3 features are extracted (*noun means in singular form*)

4. Advanced document indexing
4.3. Information extraction patterns

- [Furnkranz98]
  - The representation is evaluated on a Web categorization task (university pages classified as STUDENT, FACULTY, STAFF, DEPARTMENT, etc.
  - A Naive Bayes (NB) classifier and Ripper used
  - Results (words vs. words+phrases) are mixed
    - Accuracy improved for NB and not for Ripper
    - Precision at low recall highly improved
    - Some phrasal features are *highly predictive* for certain classes, but in general have *low coverage*

4. Advanced document indexing
4.4. Using WordNet

- Using WordNet for ATC
  - See e.g. [Buenaga00, Fukumoto01, Junker97, Petridis01, Scott98, Suzuki01]
  - WordNet is a lexical database for English with
    - high coverage of English lexical items (N, V, Adj, Adb)
    - information about lexical and semantic relations including
      - Synonymy ("car", "automobile")
      - Hyponymy – *a kind of* ("ambulance", "car")
      - Meronymy – *has part* ("car", "accelerator")
      - Etc.

4. Advanced document indexing
4.4. Using WordNet

- WordNet's organization
  - The basic unit is the synset = synonym set
  - A synset is equivalent to a concept
  - E.g. Senses of "car" (synsets to which "car" belongs)
    - {car, auto, automobile, machine, motorcar}
    - {car, railcar, railway car, railroad car}
    - {cable car, car}
    - {car, gondola}
    - {car, elevator car}

4. Advanced document indexing
4.4. Using WordNet

- WordNet's organization
  - Separated tables (files) for syntactic categories (N, V, Adj, Adb)
  - Links from words to synsets, and between synsets (representing semantic relations)
    - {person, individual, someone, somebody, mortal, human, soul}
      - *a kind of* {organism, being}
        - *a kind of* {living thing, animate thing}
          - *a kind of* {object, physical object}
            - *a kind of* {entity, physical thing}

4. Advanced document indexing
4.4. Using WordNet

- WordNet is useful for IR
  - Indexing with synsets has proven effective [Gonzalo98]
  - It improves recall because involves mapping synonyms into the same indexing object
  - It improves precision because only relevant senses are considered
    - E.g. A query for “jaguar” in the car sense causes retrieving only documents with *this word in this sense*

4. Advanced document indexing
4.4. Using WordNet

- Concept vs. sense indexing (with WordNet)
  - In concept indexing, the features are the concepts (e.g. the full synset {cable car, car})
  - In sense indexing, the features are words tagged with senses (e.g. car_N_sn3 meaning the word “car” as noun, in its third sense)
    - In this case, synonymy relation is lost, with a decrease of recall

      e.g. car_N_sn3 ≠ cable_car_N_sn1

4. Advanced document indexing
4.4. Using WordNet

- Concept indexing with WordNet
  - [Scott98, Scott99] ↓↑
    - Using synsets and hypernyms with Ripper
    - Fail because they do not perform WSD
  - [Junker97] ↓↓
    - Using synsets and hypernyms as generalization operators in a specialized rule learner
    - Fail because the proposed learning method gets *lost in the hypothesis space*

4. Advanced document indexing
4.4. Using WordNet

- Concept indexing with WordNet
  - [Petridis01] ↓↑
    - Perfect WSD (using Semcor for genre detection) with a new Neural Network algorithm
    - Senses marginally improve effectiveness
  - [Liu01] ↓↑
    - Presented a Semantic Perceptron Network (trainable semantic network) with cooccurrence, and WordNet based correlation metrics for links
    - As often, slight improvement on less populated categories

4. Advanced document indexing
4.4. Using WordNet

- Concept indexing with WordNet
  – [Fukumoto01] ↓↑
    - Sysnets and (limited) hypernyms for SVM, no WSD
    - Improvement on less populated categories
  – In general
    - Given that there is not a reliable WSD algorithm for (fine-grained) WordNet senses, current approaches do not perform WSD
    - Improvements in those categories less available information
    - *But I believe that full, perfect WSD is not required*

4. Advanced document indexing
4.4. Using WordNet

- Query expansion with WordNet
  – Often, highly relevant names are available for categories (ARTS, WHEAT, etc.)
  – This information, enriched with synonymy and WSD, has been used for ATC with
    - linear classifiers [Buenaga00, Gomez02b]
    - semi-supervised learning [Benkhalifa01]
  – Small to medium improvements

5. Task oriented features

5. Task oriented features

- In a number of TC tasks, features for learning are also stylometric or structural
  - Language identification (e.g. [Cavnar94, Sibun96, Teahan00])
  - Genre identification (e.g. [Copeck00, Finn02, Karlgren94, Kessler97, Stamatatos00, Teahan00])
  - Authorship attribution (e.g. [DeVel01, Kindermann00, Stamatatos00, Teahan00])
  - Plagiarism detection (see the survey [Clough00])
  - Spam detection ([Gomez00, Sahami98b])
  - Pornography detection
- We are concerned with easy to compute features

5. Task oriented features

- Language identification [Cavnar94]
  - Character n-grams (n=1..5)
  - Zipf's law and "out-of-place" similarity metric between distributions (made of 300 top n-grams)
  - Language identification effectiveness
    - 99,8% accuracy
  - Also ATC robust to typographic errors
    - 80% thematic newsgroup classification

5. Task oriented features

- Genre identification [Finn02]
  - Identify the degree to which a text is subjective (express author's opinions instead of facts)
  - C4.5 on bag of words (BW), POS tags freq. and 76 hand crafted (HC) features as
    - Counts of certain stop words
    - Counts of various punctuation symbols
    - Average sentence length
    - Number of long words
    - Keywords expressing subjectivity
  - Effectiveness
    - In a single domain HC > BW > POS
    - In domain transfer POS > HC > BW

5. Task oriented features

- Genre identification [Kessler97]
  - Learning algorithms are logistic regression and neural networks
  - Features include
    - Lexical
      - Terms of address (Mr.)
      - Latinate affixes
      - Words in dates
    - Character
      - Counts of question marks
      - Counts of exclamation marks
      - Counts of capitalized and hyphenated words
      - Counts of acronyms

5. Task oriented features

- Genre identification [Kessler97]
  - Features also include
    - Derivative
      - Normalized ratios of
        - » Average sentence length
        - » Average word length
        - » Words per type
      - Variations
        - » Standard deviation is sentence length
  - Effectiveness is reasonable

5. Task oriented features

- Authorship attribution [DeVel01]
  - On email for forensic investigation
  - SVMs on 170 features which include (being M the number of words and V the number of distinct words)
    - Stylistic (sample)
      - Number of blank lines/total number of lines
      - Average sentence length
      - Average word length (number of characters)
      - Vocabulary richness i.e., V=M
      - Function word frequency distribution (122 features)
      - Total number of short words/M
      - Word length frequency distribution/M (30 features)

5. Task oriented features

- Authorship attribution [DeVel01]
  - More features
    - Structural
      - Has a greeting acknowledgment
      - Uses a farewell acknowledgment
      - Contains signature text
      - Number of attachments
      - Position of requoted text within e-mail body
      - HTML tag frequency distribution/total number of HTML tags (16 features)
  - Promising accuracy for *across* and *multi-topic* author detection

5. Task oriented features

- Spam detection [Sahami98b]
  - A Naive Bayes classifier trained on stemmed words and
    - 35 hand crafted phrases from texts (“only $”, “be over 21”)
    - Domain of sender address
    - The name of sender is resolved by the email client
    - Received from a mailing list
    - Time of reception
    - Has attached files
    - Percentage of non-alphanumeric characters in subject
    - About 20 like these latter
  - Specially the latter features (not phrases) greatly increase performance reaching 96-100% precision and recall levels

5. Task oriented features

- Spam detection [Gomez00]
  - Features (9) regarded as heuristics are
    - Percentages of special characters “;”, “(“, ”[“, “!”, “$”, “#”
    - Frequencies of capital letters
  - Several learning methods (Naive Bayes, *k* Nearest Neighbors, C4.5, PART - rules)
  - With PART (best), heuristics clearly improve over word stems

5. Task oriented features

- Pornography detection (POESIA [Gomez02c])
  - We need to get more semantics
    - Metaphoric meaning of words like “screw” for erotic tales
  - Promising features include e.g.
    - Named entities (“nude pictures of <person>”)
    - Keyphrases (“be over 21”)
    - Riloff’s like syntactic signatures (“be over <number>”)
  - We expect combination of knowledge sources (images, JavaScript code analysis, etc) will improve text-based methods

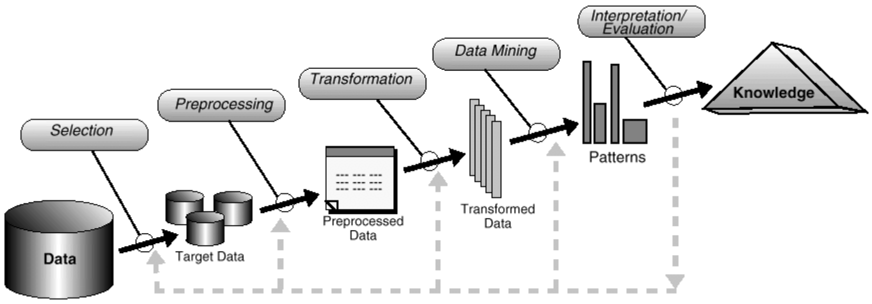
6. Summary

- General IR-ML approach works well for thematic ATC
- Features are more and more semantic

  Characters → character n-grams → word stems → phrases → syntactic patterns → concepts
- Stylistic and structural features work well for a range of useful applications
- In a real world application, approach as Knowledge Discovery in (Text) Databases

6. Summary

- The standard KDD process (borrowed from [Fayyad96])



6. Summary

1. Build or get a representative corpus
2. Label it
3. Define features
4. Represent documents
5. Learn and analyze
6. Go to 3 until accuracy is acceptable

(*first features to test: stemmed words*)

Text Representation
for Automatic Text Categorization
References

José María Gómez Hidalgo
Departamento de Inteligencia Artificial
(*Department of Artificial Intelligence*)
Universidad Europea de Madrid
28670 - Villaviciosa de Odón, Madrid, Spain
Phone: +34 91 211 56 70
jmgomez@dinar.esi.uem.es
http://www.esi.uem.es/~jmgomez/

Abstract

This is the list of references used in the tutorial titled "Text Representation for Automatic Text Categorization" at the 11th Conference of the European Chapter of the Association for Computational Linguistics, at Budapest, Hungary.

References

[Adams01] Katherine C. Adams. Representing knowledge in enterprise portals. *KMWorld Magazine*, 10(5), 2001.

[Apte94] Chidanand Apté, Fred J. Damerau, and Sholom M. Weiss. Automated learning of decision rules for text categorization. *ACM Transactions on Information Systems*, 12(3):233–251, 1994.

[Belkin92] N.J. Belkin and W.B Croft. Information filtering and information retrieval: Two sides of the same coin? *Communications of the ACM*, 35(12):29–38, 1992.

[Benkhalifa01] Mohammed Benkhalifa, Abdelhak Mouradi, and Houssaine Bouyakhf. Integrating external knowledge to supplement training data in semi-supervised learning for text categorization. *Information Retrieval*, 4(2):91–113, 2001.

[Buenaga00] Manuel de Buenaga Rodríguez, José M. Gómez Hidalgo, and Belén Díaz Agudo. Using wordnet to complement training information in text categorization. In N. Nicolov and R. Mitkov, editors, *Recent Advances in Natural Language Processing II: Selected Papers from RANLP'97*, volume 189 of *Current Issues in Linguistic Theory (CILT)*, pages 353–364. John Benjamins, 2000.

[Caropreso01] Maria Fernanda Caropreso, Stan Matwin, and Fabrizio Sebastiani. A learner-independent evaluation of the usefulness of statistical phrases for automated text categorization. In Amita G. Chin, editor, *Text Databases and Document Management: Theory and Practice*, pages 78–102. Idea Group Publishing, Hershey, US, 2001.

[Cavnar94] William B. Cavnar and John M. Trenkle. N-gram-based text categorization. In *Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval*, pages 161–175, Las Vegas, US, 1994.

[Chen00] Hao Chen and Susan T. Dumais. Bringing order to the Web: automatically categorizing search results. In *Proceedings of CHI-00, ACM International Conference on Human Factors in Computing Systems*, pages 145–152, Den Haag, NL, 2000. ACM Press, New York, US.

[Church88] K. Church. A stochastic parts program and noun phrase parser for unrestricted text. In *Proceedings of Second Conference on Applied Natural Language Processing (ANLP'88)*, 1988.

[Clough00] Paul Clough. Plagiarism in natural and programming languages: An overview of current tools and technologies. Technical Report CS-00-05, Department of Computer Science, The University of Sheffield, 2000.

[Cohen95] William W. Cohen. Learning to classify English text with ILP methods. In Luc De Raedt, editor, *Advances in inductive logic programming*, pages 124–143. IOS Press, Amsterdam, NL, 1995.

[Cohen98] William W. Cohen and Haym Hirsh. Joins that generalize: text classification using WHIRL. In Rakesh Agrawal, Paul E. Stolorz, and Gregory Piatetsky-Shapiro, editors, *Proceedings of KDD-98, 4th International Conference on Knowledge Discovery and Data Mining*, pages 169–173, New York, US, 1998. AAAI Press, Menlo Park, US.

[Copeck00] Terry Copeck, Ken Barker, Sylvain Delisle, and Stan Szpakowicz. Automating the measurement of linguistic features to help classify texts as technical. In *Proceedings of TALN 2000*, pages 101–110, 2000.

[Dagan97] Ido Dagan, Yael Karov, and Dan Roth. Mistake-driven learning in text categorization. In Claire Cardie and Ralph Weischedel, editors, *Proceedings of EMNLP-97, 2nd Conference on Empirical Methods in Natural Language Processing*, pages 55–63, Providence, US, 1997. Association for Computational Linguistics, Morristown, US.

[DeVel01] Olivier Y. de Vel, A. Anderson, M. Corney, and George M. Mohay. Mining email content for author identification forensics. *SIGMOD Record*, 30(4):55–64, 2001.

[Diaz01] A. Díaz Esteban, M. Maña López, M. de Buenaga Rodríguez, and J.M. Gómez Hidalgo. Using linear classifiers in the integration of user modeling and text content analysis in the personalization of a web-based spanish news service. In *Proceedings of the Workshop on Machine Learning, Information Retrieval and User Modeling, 8th International Conference on User Modeling*, 2001.

[Dumais92] S. Dumais and J. Nielsen. Automating the assignment of submitted manuscripts to reviewers. In *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1992.

[Fagan87] Joel L. Fagan. *Experiments in automatic phrase indexing for document retrieval: a comparison of syntactic and non-syntactic methods*. PhD thesis, Department of Computer Science, Cornell University, Ithaca, US, 1987.

[Fagan89] Joel L. Fagan. The effectiveness of a nonsyntactic approach to automatic phrase indexing for document retrieval. *Journal of the American Society for Information Science*, 40(2):115–132, 1989.

[Fayyad96] Usama Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth. Knowledge discovery and data mining: Towards an unifying framework. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*. AAAI Press, 1996.

[Finn02] Aidan Finn, Nicholas Kushmerick, and Barry Smyth. Genre classification and domain transfer for information filtering. In Fabio Crestani, Mark Girolami, and Cornelis J. van Rijsbergen, editors, *Proceedings of ECIR-02, 24th European Colloquium on Information Retrieval Research*, pages 353–362, Glasgow, UK, 2002. Springer Verlag, Heidelberg, DE. Published in the "Lecture Notes in Computer Science" series, number 2291.

[Forsyth99] Richard S. Forsyth. New directions in text categorization. In Alex Gammerman, editor, *Causal models and intelligent data management*, pages 151–185. Springer Verlag, Heidelberg, DE, 1999.

[Fukumoto01] Fumiyo Fukumoto and Yoshimi Suzuki. Learning lexical representation for text categorization. In *Proceedings of the NAACL 2001 Workshop on WordNet and Other Lexical Resources*, 2001.

[Furnkranz98] J. Furnkranz, T. Mitchell, and E. Riloff. A case study in using linguistic phrases for text categorization on the WWW. In *AAAI/ICML Workshop on Learning for Text Categorization*, 1998.

[Giraldez02] Ignacio Giráldez, Enrique Puertas, José María Gómez, Raúl Murciano, and Inmaculada Chacón. HERMES: Intelligent multilingual news filtering based on language engineering for advanced user profiling. In *Multilingual Information Access and Natural Language Processing Workshop Proceedings*, pages 81–88, 2002.

[Gomez00] José M. Gómez Hidalgo, M. Maña López, and E. Puertas Sanz. Combining text and heuristics for cost-sensitive spam filtering. In *Proceedings of the Fourth Computational Natural Language Learning Workshop, CoNLL-2000*. Association for Computational Linguistics, 2000.

[Gomez02] José María Gomez Hidalgo. Evaluating cost-sensitive unsolicited bulk email categorization. In *Proceedings of SAC-02, 17th ACM Symposium on Applied Computing*, pages 615–620, Madrid, ES, 2002.

[Gomez02b] Jose M. Gómez Hidalgo, Manuel de Buenaga Rodríguez, Luis A. Ureña López, Maria T. Martín Valdivia, and Manuel García Vega. Integrating lexical knowledge in learning-based text categorization. In *Proceedings of JADT-02, 6th International Conference on the Statistical Analysis of Textual Data*, St-Malo, FR, 2002.

[Gomez02c] José María Gómez Hidalgo, Manuel de Buenaga Rodríguez, Francisco Carrero García, and Enrique Puertas Sanz. Text filtering at POESIA: A new internet content filtering tool for educational environments. *Procesamiento del Lenguaje Natural*, 29:291–292, 2002.

[Gonzalo98] Julio Gonzalo, Felisa Verdejo, Irina Chugur, and Juan Cigarran. Indexing with WordNet synsets can improve text retrieval. In *Proceedings of the COLING/ACL Workshop on Usage of WordNet in Natural Language Processing Systems*, 1998.

[Hayes90] Philip J. Hayes and Steven P. Weinstein. CONSTRUE/TIS: a system for content-based indexing of a database of news stories. In Alain Rappaport and Reid Smith, editors, *Proceedings of IAAI-90, 2nd Conference on Innovative Applications of Artificial Intelligence*, pages 49–66. AAAI Press, Menlo Park, US, 1990.

[Hearst94] Marti Hearst. Using categories to provide context for full-text retrieval results. In *Proceedings of RIAO, Intelligent Multimedia Information Retrieval Systems and Management*, 1994.

[Hearst99] Marti A. Hearst. Untangling text data mining. In *Proceedings of ACL'99: the 37th Annual Meeting of the Association for Computational Linguistics*, 1999.

[Joachims98] Thorsten Joachims. Text categorization with support vector machines: learning with many relevant features. In Claire Nédellec and Céline Rouveirol, editors, *Proceedings of ECML-98, 10th European Conference on Machine Learning*, pages 137–142, Chemnitz, DE, 1998. Springer Verlag, Heidelberg, DE. Published in the "Lecture Notes in Computer Science" series, number 1398.

[Junker97] Markus Junker and Andreas Abecker. Exploiting thesaurus knowledge in rule induction for text classification. In Ruslan Milkov, Nicolas Nicolov, and Nilokai Nikolov, editors, *Proceedings of RANLP-97, 2nd International Conference on Recent Advances in Natural Language Processing*, pages 202–207, Tzigov Chark, BL, 1997.

[Karlgren94] Jussi Karlgren and Douglass Cutting. Recognizing text genres with simple metrics using discriminant analysis. In *Proceedings of COLING 94*, 1994.

[Kessler97] Brett Kessler, Geoff Nunberg, and Hinrich Schütze. Automatic detection of text genre. In Philip R. Cohen and Wolfgang Wahlster, editors, *Proceedings of ACL-97, 35th Annual Meeting of the Association for Computational Linguistics*, pages 32–38, Madrid, ES, 1997. Morgan Kaufmann Publishers, San Francisco, US.

[Kindermann00] Jörg Kindermann, Joachim Diederich, Edda Leopold, and Gerhard Paaß. Authorship attribution with support vector machines. In *The Learning Workshop*, 2000.

[Labrou99] Yannis Labrou and Tim Finin. YAHOO! as an ontology: using YAHOO! categories to describe documents. In *Proceedings of CIKM-99, 8th ACM International Conference on Information and Knowledge Management*, pages 180–187, Kansas City, US, 1999. ACM Press, New York, US.

[Larkey96] Leah S. Larkey and W. Bruce Croft. Combining classifiers in text categorization. In Hans-Peter Frei, Donna Harman, Peter Schäuble, and Ross Wilkinson, editors, *Proceedings of SIGIR-96, 19th ACM International Conference on Research and Development in Information Retrieval*, pages 289–297, Zürich, CH, 1996. ACM Press, New York, US.

[Larkey98] Leah S. Larkey. Automatic essay grading using text categorization techniques. In W. Bruce Croft, Alistair Moffat, Cornelis J. van Rijsbergen, Ross Wilkinson, and Justin Zobel, editors, *Proceedings of SIGIR-98, 21st ACM International Conference on Research and Development in Information Retrieval*, pages 90–95, Melbourne, AU, 1998. ACM Press, New York, US.

[Larkey99] Leah S. Larkey. A patent search and classification system. In Edward A. Fox and Neil Rowe, editors, *Proceedings of DL-99, 4th ACM Conference on Digital Libraries*, pages 179–187, Berkeley, US, 1999. ACM Press, New York, US.

[Lewis92] David D. Lewis. *Representation and learning in information retrieval*. PhD thesis, Department of Computer Science, University of Massachusetts, Amherst, US, 1992.

[Lewis92b] David D. Lewis. An evaluation of phrasal and clustered representations on a text categorization task. In Nicholas J. Belkin, Peter Ingwersen, and Annelise Mark Pejtersen, editors, *Proceedings of SIGIR-92, 15th ACM International Conference on Research and Development in Information Retrieval*, pages 37–50, Kobenhavn, DK, 1992. ACM Press, New York, US.

[Liu01] Jimin Liu and Tat-Seng Chua. Building semantic perceptron net for topic spotting. In *Proceedings of 37th Meeting of Association of Computational Linguistics (ACL2001)*, 2001.

[McCallum98] Andrew K. McCallum and Kamal Nigam. Employing EM in pool-based active learning for text classification. In Jude W. Shavlik, editor, *Proceedings of ICML-98, 15th International Conference on Machine Learning*, pages 350–358, Madison, US, 1998. Morgan Kaufmann Publishers, San Francisco, US.

[Mladenic98] Dunja Mladenić. Turning YAHOO! into an automatic Web page classifier. In Henri Prade, editor, *Proceedings of ECAI-98, 13th European Conference on Artificial Intelligence*, pages 473–474, Brighton, UK, 1998. John Wiley and Sons, Chichester, UK.

[Mladenic98b] Dunja Mladenić and Marko Grobelnik. Word sequences as features in text-learning. In *Proceedings of ERK-98, the Seventh Electrotechnical and Computer Science Conference*, pages 145–148, Ljubljana, SL, 1998.

[Mladenic98c] Dunja Mladenić. *Machine Learning on non-homogeneous, distributed text data*. PhD thesis, J. Stefan Institute, University of Ljubljana, Ljubljana, SL, 1998.

[Petridis01] Vassilios Petridis, Vassilis G. Kaburlasos, Pavlina Fragkou, and Athanasios Kehagias. Text classification using the σ-FLNMAP neural network. In *Proceedings of the 2001 International Joint Conference on Neural Networks (IJCNN2001)*, 2001.

[Porter80] Martin F. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

[Riloff94] Ellen Riloff and Wendy Lehnert. Information extraction as a basis for high-precision text classification. *ACM Transactions on Information Systems*, 12(3):296–333, 1994.

[Riloff95] Ellen Riloff. Little words can make a big difference for text classification. In Edward A. Fox, Peter Ingwersen, and Raya Fidel, editors, *Proceedings of SIGIR-95, 18th ACM International Conference on Research and Development in Information Retrieval*, pages 130–136, Seattle, US, 1995. ACM Press, New York, US.

[Riloff96] Ellen Riloff. Using learned extraction patterns for text classification. In Stefan Wermter, Ellen Riloff, and Gabriele Scheler, editors, *Connectionist, statistical, and symbolic approaches to learning for natural language processing*, pages 275–289. Springer Verlag, Heidelberg, DE, 1996. Published in the "Lecture Notes in Computer Science" series, number 1040.

[Sahami98] Mehran Sahami, Salim Yusufali, and Michelle Q. Baldonado. SONIA: a service for organizing networked information autonomously. In Ian Witten, Rob Akscyn, and Frank M. Shipman, editors, *Proceedings of DL-98, 3rd ACM Conference on Digital Libraries*, pages 200–209, Pittsburgh, US, 1998. ACM Press, New York, US.

[Sahami98b] Mehran Sahami, Susan Dumais, David Heckerman, and Eric Horvitz. A bayesian approach to filtering junk e-mail. In *Learning for Text Categorization: Papers from the 1998 Workshop*, Madison, Wisconsin, 1998. AAAI Technical Report WS-98-05.

[Salton89] Gerard Salton. *Automatic text processing: the transformation, analysis, and retrieval of information by computer*. Addison Wesley, 1989.

[Scott98] Sam Scott. Feature engineering for a symbolic approach to text classification. Master's thesis, Computer Science Department, University of Ottawa, Ottawa, CA, 1998.

[Scott99] Sam Scott and Stan Matwin. Feature engineering for text classification. In Ivan Bratko and Saso Dzeroski, editors, *Proceedings of ICML-99, 16th International Conference on Machine Learning*, pages 379–388, Bled, SL, 1999. Morgan Kaufmann Publishers, San Francisco, US.

[Sebastiani02] Fabrizio Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, 2002.

[Segal99] R. Segal and J. Kephart. MailCat: An intelligent assistant for organizing e-mail. In *Proceedings of the Third International Conference on Autonomous Agents*, 1999.

[Segal00] R. Segal and J. Kephart. Incremental learning in SwiftFile. In *Proceedings of the Seventh International Conference on Machine Learning*, 2000.

[Sibun96] Penelope Sibun and Jeffrey C. Reynar. Language identification: Examining the issues. In *5th Symposium on Document Analysis and Information Retrieval*, pages 125–135, Las Vegas, Nevada, U.S.A., 1996.

[SparckJones95] Karen Sparck Jones and Julia Rose Galliers. *Evaluating Natural Language Processing Systems : An Analysis and Review*. Lecture Notes in Computer Science – 1083 Lecture Notes in Artificial Intelligence. Springer, Berlin, 1995.

[Stamatatos00] Efstathios Stamatatos, Nikos Fakotakis, and George Kokkinakis. Automatic text categorization in terms of genre and author. *Computational Linguistics*, 26(4):471–495, 2000.

[Suzuki01] Yoshimi Suzuki, Fumiyo Fukumoto, and Yoshihiro Sekiguchi. Event tracking using WordNet meronyms. In *Proceedings of the NAACL 2001 Workshop on WordNet and Other Lexical Resources*, 2001.

[Teahan00] William J. Teahan. Text classification and segmentation using minimum cross-entropy. In *Proceeding of RIAO-00, 6th International Conference "Recherche d'Information Assistee par Ordinateur"*, Paris, FR, 2000.

[Witten99] Ian H. Witten and Eibe Frank. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann, 1999.

[Yang97] Y. Yang and J.O. Pedersen. A comparative study on feature selection in text categorization. In *Proceedings of ICML-97, 14th International Conference on Machine Learning*, 1997.

[Yang99] Yiming Yang and Xin Liu. A re-examination of text categorization methods. In Marti A. Hearst, Fredric Gey, and Richard Tong, editors, *Proceedings of SIGIR-99, 22nd ACM International Conference on Research and Development in Information Retrieval*, pages 42–49, Berkeley, US, 1999. ACM Press, New York, US.

[Zelikovitz01] Sarah Zelikovitz and Haym Hirsh. Using LSI for text classification in the presence of background text. In Henrique Paques, Ling Liu, and David Grossman, editors, *Proceedings of CIKM-01, 10th ACM International Conference on Information and Knowledge Management*, pages 113–118, Atlanta, US, 2001. ACM Press, New York, US.