Special Topics in Text Mining

Manuel Montes y Gómez

http://ccc.inaoep.mx/~mmontesg/

mmontesg@inaoep.mx

Instituto Nacional de Astrofísica, Óptica y Electrónica, Mexico.

Beyond the BoW representation

Agenda of the section

- Limitations of the BOW representation
- Some alternative features:
  - Word sequences
  - Linguistic features
  - Word senses as features
  - Concept-based representations
    - Distributional representations (DOR and TCOR)
    - Random indexing
    - Other representations

Bag-of-Words representation

- Very common because its **simplicity** and **efficiency**.
- Under this scheme, documents are represented by collections of terms, each term being an independent feature.
  - Word order is not capture by this representation
  - Semantic information is omitted
  - There is no attempt for understanding documents' content

The BoW representation

Vocabulary from the collection
(*set of different words*)

	t_1	t_1	…	t_n
d_1				
d_2				
:		$w_{i,j}$		
d_m				

All documents
(*one vector per document*)

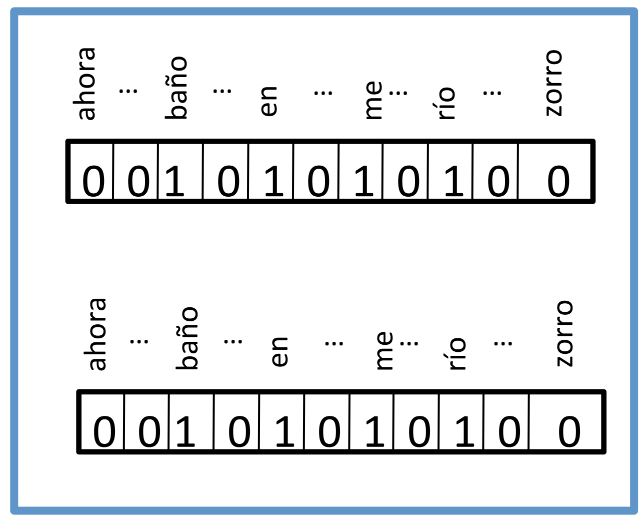
Weight indicating the contribution of word *j* in document *i*.

Main problems

- A document is represented by the set of terms that appear in it
- By definition, BOW is an orderless representation

Yo me rio en el baño
(*I am laughing at the bathroom*)

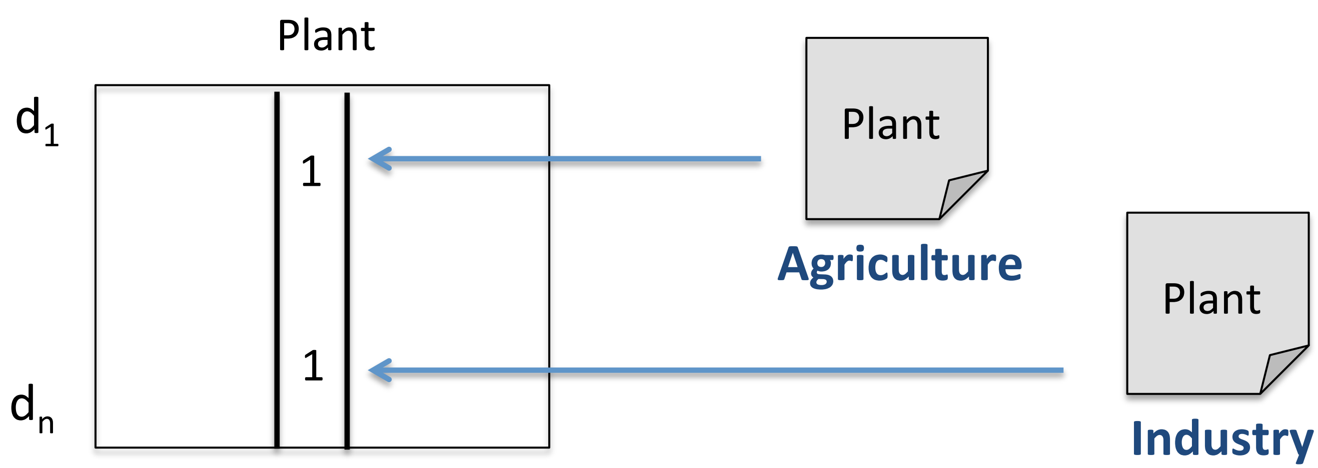
Yo me baño en el río
(*I am taking a shower at the river*)



Same BoW representation different meaning

Main problems

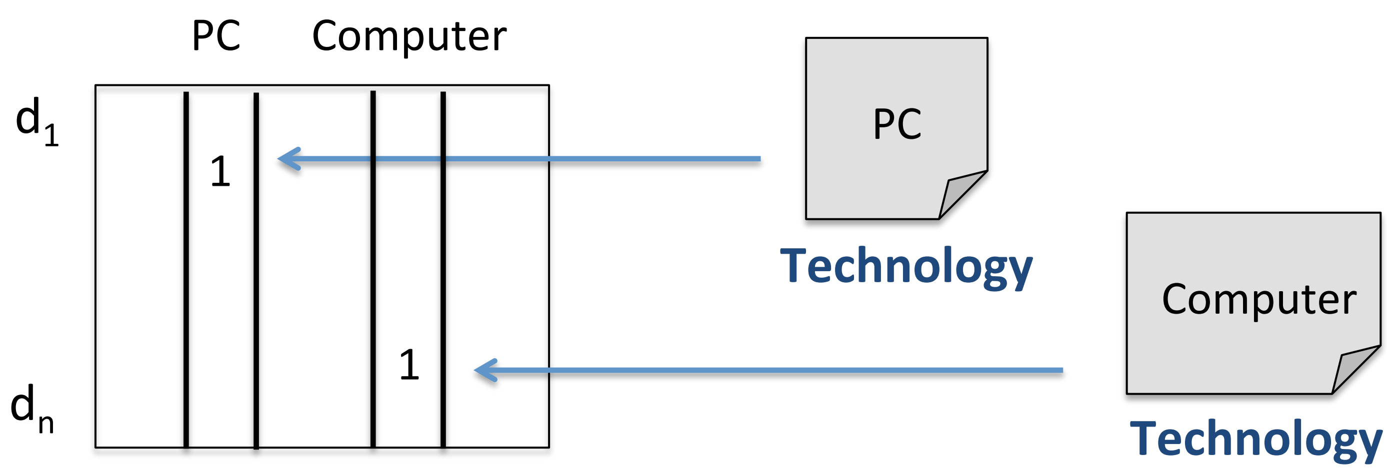
- BoW ignores all semantic information; it simply looks at the surface word forms
  - Polysemy and synonymy are big problems



Polysemy introduces noise into the BOW representation

Main problems

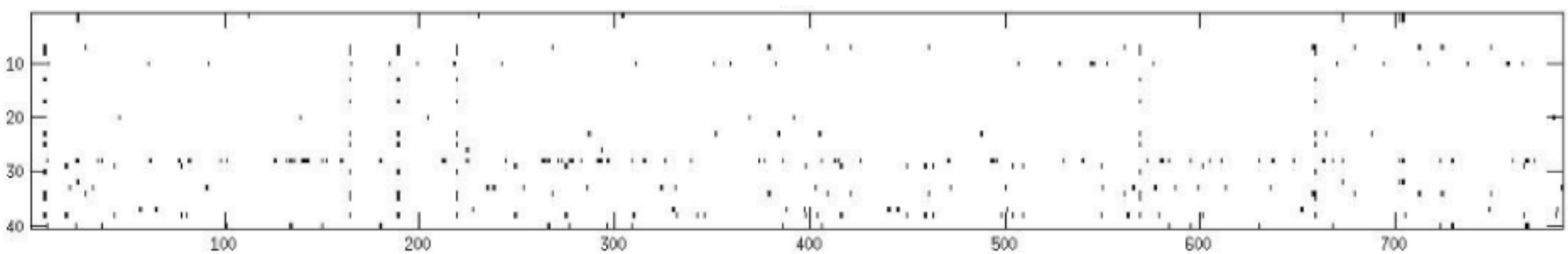
- BoW ignores all semantic information; it simply looks at the surface word forms
  - Polysemy and synonymy are big problems



Synonymy splits the evidence in the BOW representation

Main problems

- BoW tends to produce sparse representations, since terms commonly occur in just a small subset of the documents
  – This problem is amplified by lack of training texts and by the shortness of the documents



Very difficult to find classification patterns!

Ideas for solving these limitations?

First idea: indexing with POS tags

Whole vocabulary of the collection with POS tags

	w_1t_1	w_1t_2	Plant|NN	Plant|VB	…	w_nt_m
d_1						
d_2						
:		$w_{i,j}$				
d_m						

Weight indicating the contribution of term-pos *j* in document *i*.

Comments on this solution? Does it works?

Second idea: motivation

- Using single words as index terms generally has good exhaustivity, but poor specificity due to word ambiguity.
- Some word associations have a totally different meaning of the "sum" of the meanings of the words that compose them.
  - Hot + dog ≠ "hot dog"
- To remedy this problem: use terms more complex than single words, such as *phrases*.
  - Distinguish the two meanings by using phrasal index terms such as "bank of the Seine" and "bank of Japan"

Second idea: phrases as features

Extracted phrases from the collection

	p_1	P_2	Information retrieval	Paul McCartney	Rolling Stones	p_n
d_1						
d_2						
:		$w_{i,j}$				
d_m						

Weight indicating the contribution of phrase *j* in document *i*.

Which kind of word sequences are relevant phrases?
How to extract them?

Syntactical phrases as features

This apple pie looks good and is a real treat

- adjective-noun relation (*real-treat*)
- noun-noun relation (*apple-pie*)
- subject-verb relation (*pie-looks*)
- verb-object relation (*is-treat*)
- The complication is that they are extracted from the POS tagged text or from the *syntactic tree*.

Named entities as features

- *Proper names* in texts
  - Three universally accepted categories: **person**, **location** and **organisation**
  - Other categories: date/time expressions, measures (percent, money, weight etc), email addresses, etc.
- One problem: they can be also ambiguous!
  - George Bush: person or location?
  - Mexico: geo-political organization or location?

How to detect named entities?

N-grams as features

- N-gram is a subsequence of *n* items from a given sequence
- N-grams are <u>easily computed</u>
- Combining n-grams for different sizes produces great coverage and flexibility for the representation.
- Main problem is the <u>high dimensionality</u>.

How to select only the most useful n-grams?

Third idea: motivation

- Traditional IR/TC approaches are highly dependent on *term-matching*
- Term matching is affected by the *synonymy* and *polysemy* phenomena.
- Need to capture the concepts instead of only the words
- Solution: using word senses as features!

What is word sense?

- Word sense is one of the *meanings* of a word.
- "Words" are having different meanings based on the context of the word.
- Example:
  - We went to see a play at the theater
  - The children went out to play in the park

A computer program has no basis for knowing which one is appropriate, even if it is obvious to a human

Third idea: indexing by senses

All different word senses from the target collection

	w_{11}	w_{12}	Bank (institution)	Bank (hill)	p_{n1}	p_{nm}
d_1						
d_2						
:		$w_{i,j}$				
d_m						

Weight indicating the contribution of the word-sense *j* in document *i*.

We need to determine the sense of each word from the document collection. Hard problem!

Did they work?

- Evidence that POS info, complex nominals, and word senses do not improve TC accuracy.
  – Lack of accurate NLP tools (in many languages)
  – High computational cost in comparison with BOW
- The combination of word unigrams and bigrams tend to produce the best results.
  – Higher order n-grams are –usually– useless.

So, what else can we try? Ideas?

Alessandro Moschitti, Roberto Basili. *Complex Linguistic Features for Text Classification: A Comprehensive Study*. Lecture Notes in Computer Science Volume 2997, 2004.c

Bag-of-concepts

- Addresses the deficiencies of the BoW by considering the relations between document terms.
- BoC representations are based on the intuition that the meaning of a document can be considered as the union of the meanings of their terms.
- The meaning of terms is related to their usage; it is captured by their distributional representation

Alberto Lavelli, Fabrizio Sebastiani, and Roberto Zanoli. Distributional term representations: an experimental comparison. *Thirteenth ACM international conference on Information and knowledge management* (CIKM '04). New York, NY, USA, 2004

Document Occurrence Representation

- It is based on the idea that the semantics of a term may be view as a function of the bag of documents in which the term occurs.
  - Each document being an independent feature
- Terms are represented as vectors in the space of documents
- Two terms are related if they show similar distributions across the documents

Representation of terms

	d_1	d_2	…	d_n
t_1				
t_2				
:		$w_{i,j}$		
t_m				

Intuitions about the weights

	d_1	d_2	…	d_n
t_1				
t_2				
:		$w_{i,j}$		
t_m				

$$w_{k,j} = \underline{df(d_k,t_j)} \cdot \underline{log\frac{|T|}{N_k}}$$

$$df(d_k,t_j) = \begin{cases} 1+\log(\#(d_k,t_j)) & if(\#(d_k,t_j)>0) \\ 0 & otherwise \end{cases}$$

- DOR is a **dual version of the BoW** representation, therefore:
  - The more frequently t_i occurs in d_j, the more important is d_j for characterizing the semantics of t_i
  - The more distinct the words d_j contains, the smaller its contribution to characterizing the semantics of t_i.

Representing documents using DOR

- DOR is a word representation, not a document representation.
- Representation of documents is obtained by the weighted **sum of the vectors** from their terms.

$$d_i^{dtr} = \sum_{t_j \in d_i} \alpha_{t_j} \cdot w_{t_j}$$

Word representation
Word–Document matrix

	d_1	d_2	…	d_n
t_1				
t_2				
:		$w_{i,j}$		
t_m				

SUM

Document representation
Document–Document matrix

	d_1	d_2	…	d_n
d_1				
d_2				
:		$w_{i,j}$		
d_n				

Term CO-occurrence Representation

- In TCOR, the meaning of a term is conveyed by the terms commonly co-occurring with it; i.e. terms are represented by the terms occurring in their context
- Terms are represented as vectors in the space of terms (vocabulary of the collection)
- Two terms are related if they show similar co-ocurring distributions with the rest of the terms

Representation of terms

	t_1	t_2	…	t_m
t_1				
t_2				
:		$w_{i,j}$		
t_m				

Intuitions about the weights

	t_1	t_2	…	t_m
t_1				
t_2				
:		$w_{i,j}$		
t_m				

$$w_{k,t} = \underline{tff(t_k,t_j)} \cdot \underline{log\frac{|T|}{T_k}}$$

$$tff(t_k,t_j) = \begin{cases} 1+\log(\#(t_k,t_j)) & if(\#(t_k,t_j)>0) \\ 0 & otherwise \end{cases}$$

- TCOR is the kind of representation traditionally used in WSD, therefore:
  - The more times t_k and t_j co-occur in, the more important t_k is for characterizing the semantics of t_j
  - The more distinct words t_k co-occurs with, the smaller its contribution for characterizing the semantics of t_j.

Representing documents using TCOR

- TCOR, such as DOR, is a word representation, not a document representation.
- Representation of documents is obtained by the weighted ors from their terms.

$$d_i^{dtr} = \sum_{t_j \in d_i} \alpha_{t_j} \cdot w_{t_j}$$

Word representation
Word–Word matrix

	t_1	t_2	…	t_m
t_1				
t_2				
:				
t_m				

SUM

Document representation
Document–Word matrix

	t_1	t_2	…	t_m
d_1				
d_2				
:				
d_n				

BOW vs DOR vs TCOR

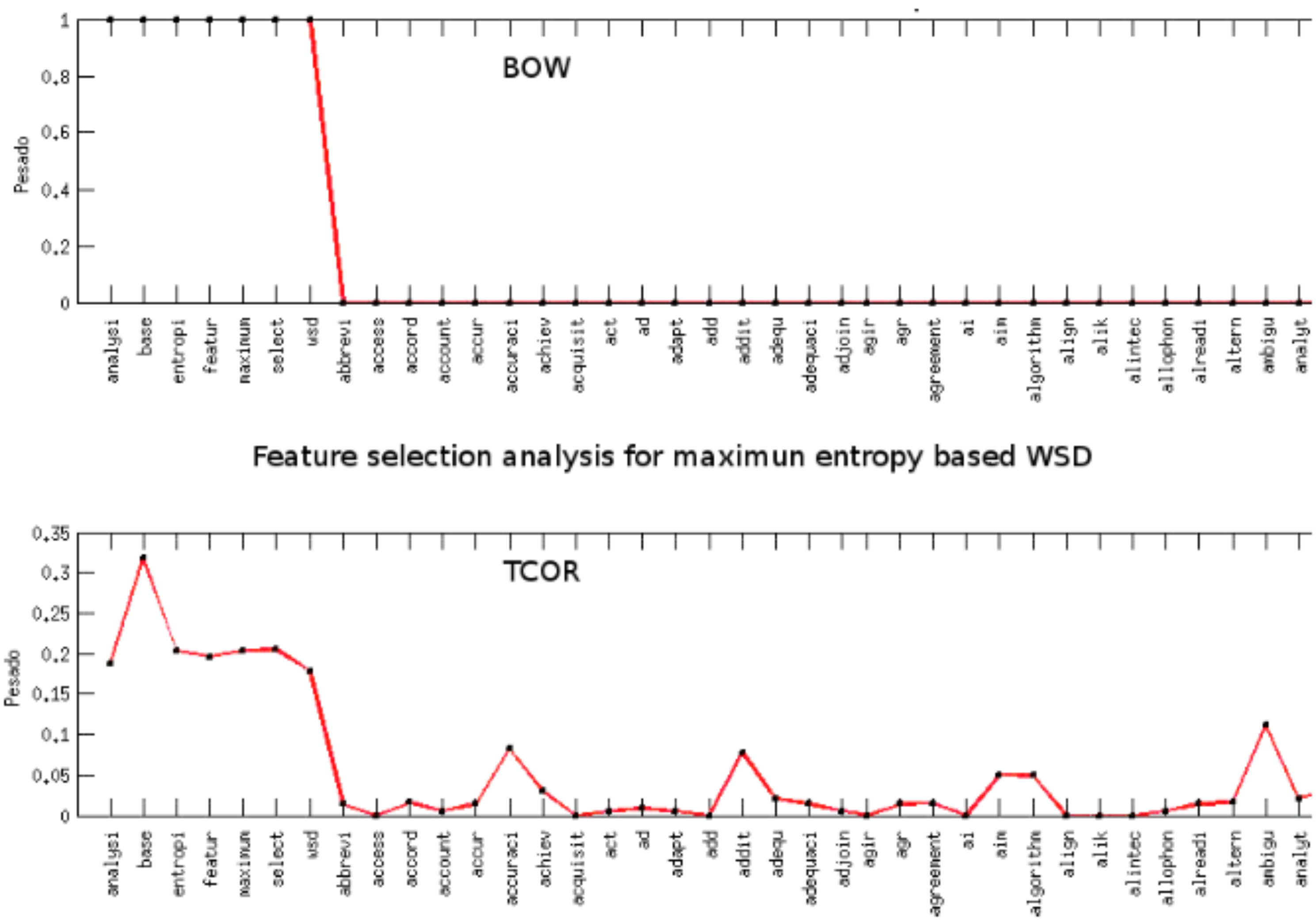
	t_1	t_2	…	t_m
d_1				
d_2				
:				
d_n				

	d_1	d_2	…	d_n
d_1				
d_2				
:		$\mathrm{w}_{i,j}$		
d_n				

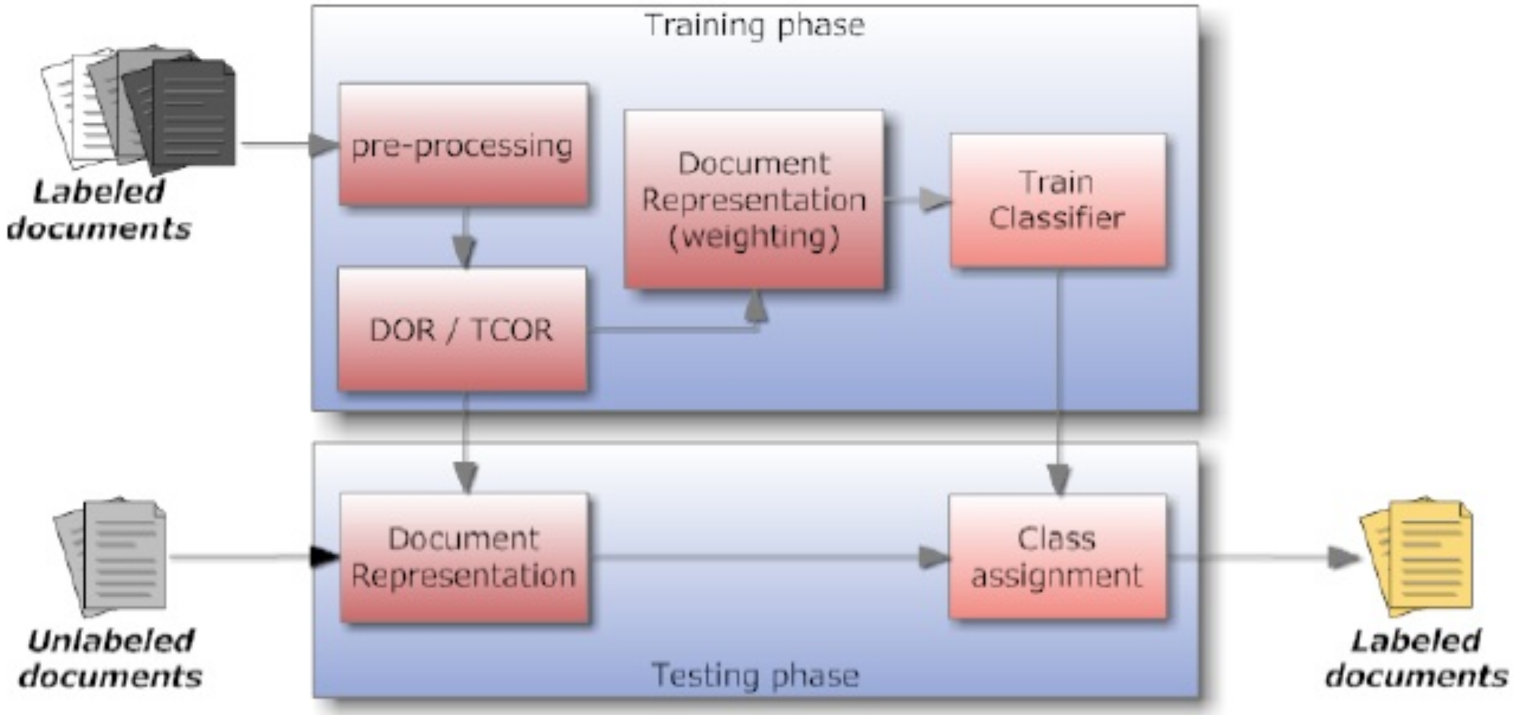
- BOW
  - High dimensionality
  - Very sparse
- DOR
  - Lower dimensionality than BOW
  - Not sparse
- TCOR
  - Same dimensionality than BOW
  - Not sparse

DOR and TCOR do a kind of expansion of the documents

TCOR representation of a paper title



DOR/TCOR for text classification



Juan Manuel Cabrera, Hugo Jair Escalante, Manuel Montes-y-Gómez. Distributional term representations for short text categorization. *14th International Conference on Intelligent Text Processing and Computational Linguistics* (CICLING 2013). Samos, Greece, 2013.

Experiments

- Short-text categorization based on distributional term representations
  - They **reduce the sparseness** of representations and alleviates, to some extent, the low frequency issue.
- Our experiments aimed to:
  - Verify the difficulties of the BoW for effectively representing the content of short-texts
  - Assess the added value offered by concept-based representations over the BoW formulation

Evaluation datasets

- We assembled two types of collections:
  - Whole documents for training and test
  - Whole documents for training and titles for test

Feature	Train	Test (DD)	Test-Reduced (DT)
Vocabulary size	14,865	8,760	3,676
Number of Documents	4,559	2,179	2,179
Average terms per document	40.9	39.2	6.6

Reuters-R8

Feature	Regular (DD)	Reduced (DT)
Vocabulary size	1136	206
Number of Documents	48	48
Average terms per doc.	60.3	5.85

EasyAbstracts

Feature	Regular (DD)	Reduced (DT)
Vocabulary size	813	180
Number of Documents	48	48
Average terms per doc.	45.06	4.8

Cicling 2002

Short-text classification with BoW

R8

	Boolean			TF			TFIDF		
	DD	DT	Decrease	DD	DT	Decrease	DD	DT	Decrease
AdaBoost	0.64	0.18	-72.74%	0.64	0.18	-72.74%	0.64	0.18	-72.74%
Knn1	0.69	0.39	-43.98%	0.47	0.34	-27.53%	0.47	0.34	-27.53%
Naive Bayes	0.87	0.66	-24.16%	0.82	0.34	-58.97%	0.82	0.34	-59.13%
RandomForest	0.80	0.54	-32.21%	0.80	0.57	-29.02%	0.82	0.74	-10.46%
SVMLineal	**0.91**	**0.83**	-7.85%	0.90	0.73	-19.29%	0.90	0.70	-22.59%

EasyAbstract

AdaBoost	0.41	0.27	-34.34%	0.40	0.25	-37.70%	0.40	0.25	-37.70%
Knn1	0.21	0.11	-46.14%	0.14	0.09	-38.74%	0.14	0.09	-38.74%
Naive Bayes	0.70	0.40	-42.89%	0.74	0.35	-53.09%	0.79	0.37	-52.93%
RandomForest	0.57	0.24	-57.82%	0.49	0.22	-54.34%	0.53	0.19	-64.01%
SVMLineal	**0.69**	**0.59**	-15.64%	0.90	0.16	-82.05%	0.85	0.30	-64.67%

CICLing

AdaBoost s	0.36	0.27	-22.76%	0.36	0.27	-22.76%	0.31	0.20	-35.32%
Knn1	0.29	0.10	-65.62%	0.14	0.16	10.62%	0.13	0.09	-31.31%
Naive Bayes	0.43	0.33	-23.50%	0.43	0.39	-10.50%	0.37	0.14	-61.30%
RandomForest	0.40	0.25	-38.01%	0.31	0.30	-1.10%	0.22	0.12	-46.91%
SVMLineal	0.45	0.35	-21.14%	**0.54**	**0.48**	-11.91%	0.21	0.14	-35.52%

Conclusions (1)

- Acceptable performance was obtained when regular-length documents were considered
  - SVM obtained the best results for most configurations of data sets and weighting schemes
- The performance of most classifiers dropped considerably when classifying short documents
  - The average decrement of accuracy was of 38.66%
- Results confirm that the BoW representation is not well suited for short-text classification

Using DOR/TCOR for short-text classification

R8

Weigth	Boolean			TF			TFIDF		
Classifiers	BOW	DOR	TCOR	BOW	DOR	TCOR	BOW	DOR	TCOR
AB	0.175	0.645	0.668	0.175	0.632	0.651	0.175	0.591	0.667
KNN	0.386	**0.899**	**0.897**	0.337	**0.908**	**0.902**	0.337	**0.746**	**0.754**
NB	0.656	**0.881**	**0.893**	0.336	**0.874**	**0.886**	0.336	**0.785**	**0.854**
RF	0.543	**0.786**	**0.774**	0.565	**0.805**	**0.823**	0.736	**0.798**	**0.819**
SVM	0.834	**0.930**	**0.891**	0.728	**0.928**	**0.901**	0.699	**0.897**	**0.784**
EasyAbstract									
AB	0.268	0.185	0.201	0.255	0.272	0.245	0.250	0.263	0.292
KNN	0.114	**0.600**	**0.482**	0.086	**0.666**	**0.712**	0.086	**0.571**	**0.541**
NB	0.402	**0.568**	**0.586**	0.345	**0.603**	**0.590**	0.370	**0.578**	**0.603**
RF	0.239	**0.495**	**0.332**	0.223	**0.507**	**0.582**	0.192	**0.588**	**0.550**
SVM	0.585	0.660	**0.639**	0.161	**0.728**	**0.733**	0.301	**0.622**	**0.589**
CICLIng2002									
AB	0.274	0.188	0.244	0.274	0.129	0.224	0.199	0.201	0.232
KNN	0.099	**0.450**	**0.395**	0.156	**0.478**	**0.399**	0.089	**0.493**	**0.44**
NB	0.332	**0.473**	0.415	0.386	**0.426**	0.471	0.143	**0.506**	**0.399**
RF	0.249	0.184	0.369	0.304	0.279	0.374	0.119	**0.418**	**0.291**
SVM	0.354	**0.526**	0.414	0.48	0.504	0.502	0.135	**0.528**	**0.442**

Conclusions (2)

- DOR and TCOR clearly outperformed BoW for most configurations.
  - In 62 out of the 90 results the improvements of DTRs over BoW were statistically significant
- In average, results obtained with DOR and TCOR were very similar.
  - DOR is advantageous over TCOR because it may result in document representations of much lower dimensionality.

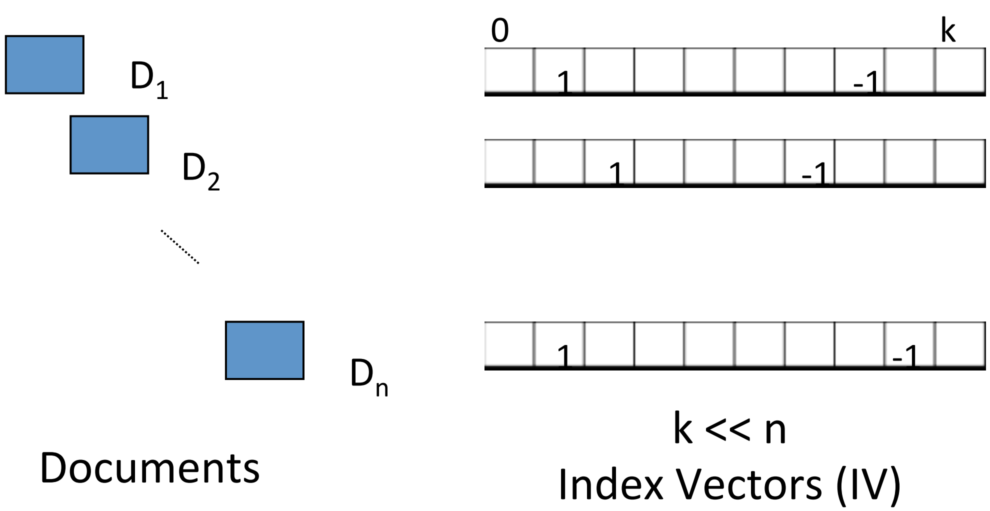
Bag of concepts by random indexing

- BoC approaches tend to be computationally expensive.
- They are based on a co-occurrence matrix of order $w \times c$; w = terms, c = contexts (terms or documents)
- Random indexing produce these context vectors in a more computationally efficient manner: the co-occurrence matrix is replaced by a context matrix of order $w \times k$, where $k << c$.

Magnus Sahlgren and Rickard Cöster. Using bag-of-concepts to improve the performance of support vector machines in text categorization. *20th international conference on Computational Linguistics* (COLING '04). Stroudsburg, PA, USA, 2004.

Random indexing procedure (1)

- *First step*: a unique random representation known as "index vector" is assigned to each context.
  - A context could be a document , paragraph or sentence
  - Vectors are filled with -1, 1 and 0s.



Random indexing procedure (2)

- *Second step*: index vectors are used to produce context vectors by scanning through the text

D$_1$: Towards an Automata Theory of Brain

D$_2$: From Automata Theory to Brain Theory

	1				-1				
1							-1		

0 k

The context vector for brain

1	1				-1		-1		

- Same idea than DOR: terms are represented by the documents they occur.
- The context vector includes information from all documents containing the term

Random indexing procedure (2)

- *Third step*: build document vectors by adding their terms' context vectors.

d_i: "From Automata Theory to Brain Theory"

CV_1 CV_2 CV_3 CV_2

d_i will be represented as the weighted sum of these vectors:

$a_1CV_1+a_2CV_2+a_3CV_3+a_2CV_2$ a_1, a_2, a_3 are idf-values

- As in DOR and TCOR, the representation of the documents is obtained by the weighted sum of the context vectors of their terms.
- It is like having a new code bar for each document which summarize all its information

Limitations of BoC representations

- BoC representations ignore the large amount of syntactic data in the documents not captured implicitly through term context co-occurrences
- Although BoC representations can successfully model some synonymy relations, since different words with similar meaning will occur in the same contexts, they cannot model polysemy.
- Solution: a representation that encodes both the semantics of documents, as well as the syntax of documents

RI with syntactic information

- Multiplicative bidding procedure:
  - For each PoS tag, generate a unique random vector for the tag of the same dimensionality as the term context vectors.
  - For each term context vector, we perform element-wise multiplication between that term's context vector and its identified PoS tag vector to obtain our combined representation for the term.
  - Finally, document vectors are created by summing the combined term vectors.

Jonathan M. Fishbein and Chris Eliasmith. Methods for augmenting semantic models with structural information for text classification. *30th European conference on Advances in information retrieval* (ECIR'08). Glasgow, UK, 2008.

An alternative procedure

- Circular convolution procedure:
  - For each PoS tag, generate a unique random vector for the tag of the same dimensionality as the term context vectors
  - For each term context vector, perform circular convolution, which binds two vectors :

term $\underline{A} = (a_0, a_1, \ldots, a_{n-1})$

tag $\underline{B} = (b_0, b_1, \ldots, b_{n-1})$

term-tag $\underline{C} = (c_0, c_1, \ldots, c_{n-1})$

$$\underline{C} = \underline{A} \otimes \underline{B}$$

$$c_j = \sum_{k=0}^{n-1} a_k b_{j-k}$$

  - Finally, document vectors are created by summing the combined term vectors

Circular convolution as binding operation

- Two properties that make it appropriate to be used as a binding operation:
  - The expected similarity between a convolution and its constituents is zero, thus **differentiating the same term acting as different parts of speech** in similar contexts.
    - Gives high importance to syntactic information
  - Similar semantic concepts (i.e., term vectors) bound to the same part-of-speech will result in similar vectors; therefore, usefully **preserving the original semantic model**.
    - Preserves semantic information

Results on text classification

- The goal of the experiment was to demonstrate that integrating PoS data to the text representation is useful for classification purposes.
- Experiments on the 20 Newsgroups corpus; a linear SVM kernel function was used; all context vectors were fixed to 512 dimensions

Syntactic Binding Method	$\mathcal{F}_1$ Score
BoC (No Binding)	56.55
Multiplicative Binding	57.48
Circular Convolution	**58.19**

Final remarks

- BoC representations constitute a viable supplement to word based representions.
- Not too much work in text classification and IR
  - Recent experiments demonstrated that TCOR, DOR and random indexing results outperform those from traditional BoW; in CLEF collections improvements have been around 7%.
- Random indexing is efficient, fast and scalable; syntactic information is easily incorporated.

Related approaches

- **Latent semantic indexing:** Concepts are derived via SVD, concepts are the *principal components* of the term-document matrix
- **Topic models:** Concepts are probability distributions over words, they can be obtained in different ways (pLSI, LDA, etc.)
- **Deep learning:** Concepts are the outputs of hierarchical neural networks that aimed to reconstruct documents (word2vec)

Explicit Semantic Analysis

- It is a representation of documents that uses a document corpus as a knowledge base.
  - Concepts explicitly defined and described by humans.
- The idea is to represent documents by their relatedness with a set of explicitly given external categories or concepts
  - Wikipedia articles are commonly used as these external categories.

Gabrilovich, E.; Markovitch, S (2006). Overcoming the brittleness bottleneck using Wikipedia: enhancing text categorization with encyclopedic knowledge. Proc. 21st National Conference on Artificial Intelligence (AAAI). pp. 1301–1306.

Using Wikipedia articles as features

Each Wikipedia paper is a feature

	w_1	w_2			W_{n-1}	W_n
d_1						
d_2						
:		$w_{i,j}$				
d_m						

Weight indicating the relation of category (article) *j* to document *i*.

Some results of ESA

Dataset	Baseline micro	Baseline macro	Wikipedia micro	Wikipedia macro	Improvement micro	Improvement macro
Reuters-21578 (10 cat.)	0.925	0.874	0.932	0.887	+0.8%	+1.5%
Reuters-21578 (90 cat.)	0.877	0.602	0.883	0.603	+0.7%	+0.2%
RCV1 Industry-16	0.642	0.595	0.645	**0.617**	+0.5%	**+3.7%**
RCV1 Industry-10A	0.421	0.335	**0.448**	**0.437**	**+6.4%**	**+30.4%**
RCV1 Industry-10B	0.489	0.528	**0.523**	**0.566**	**+7.0%**	**+7.2%**
RCV1 Industry-10C	0.443	0.414	**0.468**	**0.431**	**+5.6%**	**+4.1%**
RCV1 Industry-10D	0.587	0.466	0.595	0.459	+1.4%	-1.5%
RCV1 Industry-10E	0.648	0.605	0.641	0.612	-1.1%	+1.2%
RCV1 Topic-16	0.836	0.591	0.843	**0.661**	+0.8%	**+11.8%**
RCV1 Topic-10A	0.796	0.587	0.798	**0.682**	+0.3%	**+16.2%**
RCV1 Topic-10B	0.716	0.618	0.723	**0.656**	+1.0%	**+6.1%**
RCV1 Topic-10C	0.687	0.604	0.699	0.618	+1.7%	+2.3%
RCV1 Topic-10D	0.829	0.673	0.839	0.688	+1.2%	+2.2%
RCV1 Topic-10E	0.758	0.742	0.765	0.755	+0.9%	+1.8%
OHSUMED-10A	0.518	0.417	**0.538**	**0.492**	**+3.9%**	**+18.0%**
OHSUMED-10B	0.656	0.500	0.667	**0.534**	+1.7%	**+6.8%**
OHSUMED-10C	0.539	0.505	0.545	**0.522**	+1.1%	**+3.4%**
OHSUMED-10D	0.683	0.515	0.692	**0.546**	+1.3%	**+6.0%**
OHSUMED-10E	0.442	0.542	**0.462**	**0.575**	**+4.5%**	**+6.1%**
20NG	0.854		**0.862**		**+1.0%**	
Movies	0.813		**0.842**		**+3.6%**	

Table 1: The effect of feature generation

DATASET	Baseline micro	Baseline macro	Wikipedia micro	Wikipedia macro	Improvement micro	Improvement macro
Reuters-21578 (10 cat.)	0.868	0.774	0.877	0.793	+1.0%	+2.5%
Reuters-21578 (90 cat.)	0.793	0.479	0.803	**0.506**	+1.3%	**+5.6%**
RCV1 Industry-16	0.454	0.400	**0.481**	**0.437**	**+5.9%**	**+9.2%**
RCV1 Industry-10A	0.249	0.199	**0.293**	**0.256**	**+17.7%**	**+28.6%**
RCV1 Industry-10B	0.273	0.292	**0.337**	**0.363**	**+23.4%**	**+24.3%**
RCV1 Industry-10C	0.209	0.199	**0.294**	**0.327**	**+40.7%**	**+64.3%**
RCV1 Industry-10D	0.408	0.361	**0.452**	**0.379**	**+10.8%**	**+5.0%**
RCV1 Industry-10E	0.450	0.410	**0.474**	**0.434**	**+5.3%**	**+5.9%**
RCV1 Topic-16	0.763	0.529	0.769	0.542	+0.8%	+2.5%
RCV1 Topic-10A	0.718	0.507	0.725	**0.544**	+1.0%	**+7.3%**
RCV1 Topic-10B	0.647	0.560	0.643	0.564	-0.6%	+0.7%
RCV1 Topic-10C	0.551	0.471	**0.573**	**0.507**	**+4.0%**	**+7.6%**
RCV1 Topic-10D	0.729	0.535	0.735	**0.563**	+0.8%	**+5.2%**
RCV1 Topic-10E	0.643	0.636	0.670	0.653	+4.2%	+2.7%
OHSUMED-10A	0.302	0.221	**0.405**	**0.299**	**+34.1%**	**+35.3%**
OHSUMED-10B	0.306	0.187	**0.383**	**0.256**	**+25.2%**	**+36.9%**
OHSUMED-10C	0.441	0.296	**0.528**	**0.413**	**+19.7%**	**+39.5%**
OHSUMED-10D	0.441	0.356	**0.460**	**0.402**	**+4.3%**	**+12.9%**
OHSUMED-10E	0.164	0.206	**0.219**	**0.280**	**+33.5%**	**+35.9%**
20NG	0.699		**0.749**		**+7.1%**	

Table 2: Feature generation for short documents

Comments on ESA

- It is compared to approaches which aim at representing texts with respect to latent topics or concepts, as done in Latent Semantic Analysis.
  - However, the use of a knowledge base makes it possible to assign human-readable labels to the concepts.
- Empirical evaluation confirms that using ESA leads to substantial improvements in computing word and text relatedness.
  - ESA have improved text categorization
  - We are using ESA for author profiling tasks.

Some other (new) representations

- LOWBOW: Local Bag of Words
  - Allow to include order info into the BoW
- Concise semantic analysis
  - Represent documents in the space of categories
- Multimodal metafeatures
  - Combines different kinds (modalities) of features
  - Represents documents by their similarity with some prototypes.

These representations are going to be discussed in the Section "Authorship analysis"

References

- Ron Bekkerman, Ran El-Yaniv, Naftali Tishby, and Yoad Winter. **Distributional word clusters vs. words for text categorization**. *Journal of Machine Learning Research.* March 2003.
- Juan Manuel Cabrera, Hugo Jair Escalante, Manuel Montes-y-Gómez. **Distributional term representations for short text categorization**. *14th International Conference on Intelligent Text Processing and Computational Linguistics* (CICLING 2013). Samos, Greece, 2013.
- Jonathan M. Fishbein and Chris Eliasmith. **Methods for augmenting semantic models with structural information for text classification**. *30th European conference on Advances in information retrieval* (ECIR'08). Glasgow, UK, 2008.
- Alberto Lavelli, Fabrizio Sebastiani, and Roberto Zanoli. **Distributional term representations: an experimental comparison**. *Thirteenth ACM international conference on Information and knowledge management* (CIKM '04). New York, NY, USA, 2004.
- Magnus Sahlgren and Rickard Cöster. **Using bag-of-concepts to improve the performance of support vector machines in text categorization**. *20th international conference on Computational Linguistics* (COLING '04). Stroudsburg, PA, USA, 2004.
- Dou Shen, Jianmin Wu, Bin Cao, Jian-Tao Sun, Qiang Yang, Zheng Chen, and Ying Li. **Exploiting term relationship to boost text classification**. *18th ACM conference on Information and knowledge management* (CIKM '09). New York, NY, USA, 2009.
- Peter D. Turney and Patrick Pantel. **From Frequency to Meaning: Vector Space Models of Semantic**s. *Journal of Artificial Intelligence Research,* vol. 37, 2010.