

Learning Word-to-Concept Mappings for Automatic Text Classification

Georgiana Ifrim

Martin Theobald

Gerhard Weikum

- Introduction
- Related Work
- Probabilistic Model
- Experimental Results
- Conclusion & Future Work

□ Motivation

- Richness of features to represent documents – possible bottleneck for obtaining high accuracy in text categorization
- Try to exploit language semantics to overcome it

□ Our Contribution

- Probabilistic mapping of words to their meanings through an iterative EM process, coupled with a classifier for topic labelling
- Bootstrapping initialization heuristic for avoiding combinatorial explosion of parameter space and possible EM flaws

□ Work on spectral decomposition

- Deerwester, Dumais & Harshman, 1990 (LSA)
- Hoffman, 2001 (PLSA)
- Represent documents in a reduced space of concepts; Require specifying number of concepts/factors *a priori*

□ Feature engineering & Wordnet (WN)

- Cai & Hoffman, 2003 (PLSA + AdaBoost)
- Bloedhorn & Hotho, 2004 (explicit concepts from WN + AdaBoost)
- Scott & Matwin, 1998 (explicit concepts from WN + RIPPER)
- Enhance word feature space by concepts & plug into classifier

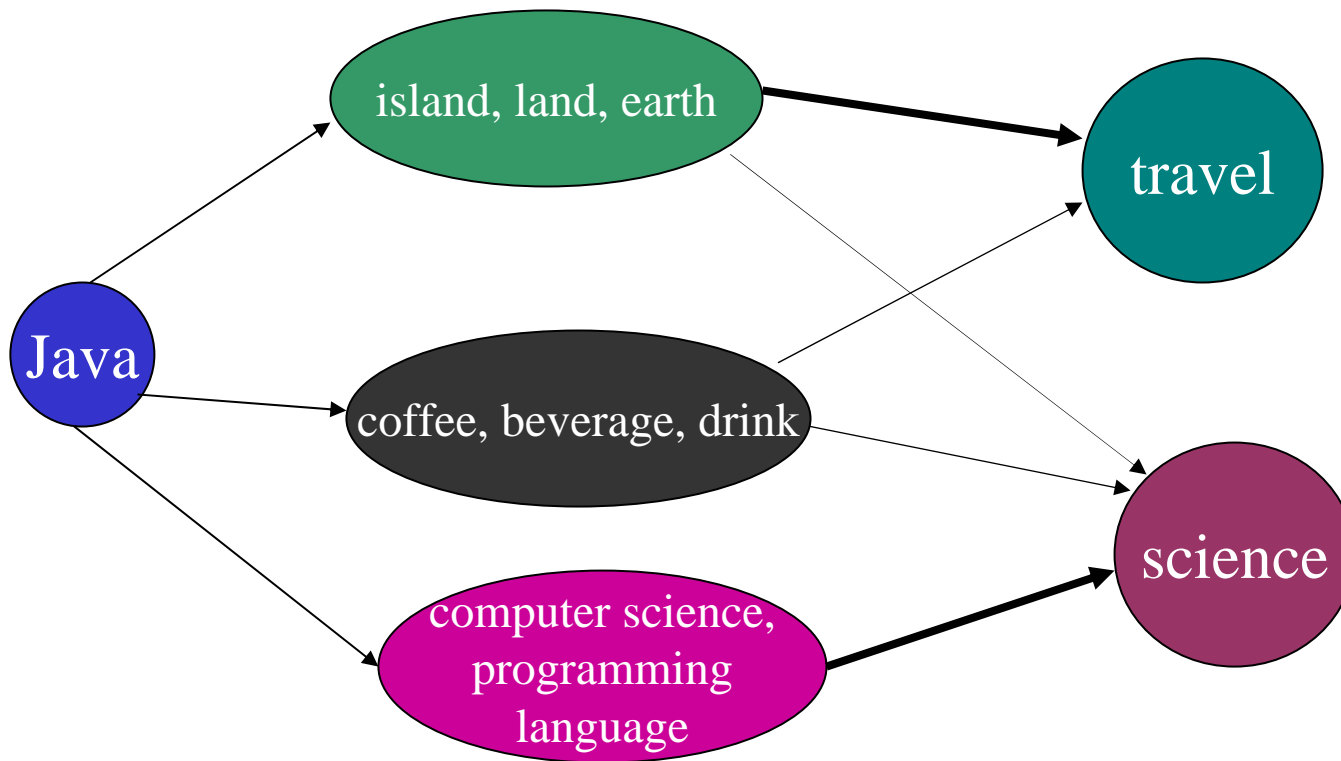
□ Given

- A data collection (Reuters-21578, Amazon)
 - A set of training documents with known topic labels and **observed** features, but **latent concepts**
- An ontology of concepts (WordNet)
 - Each concept has a set of synonyms, a short textual description and is linked by hierarchical relations

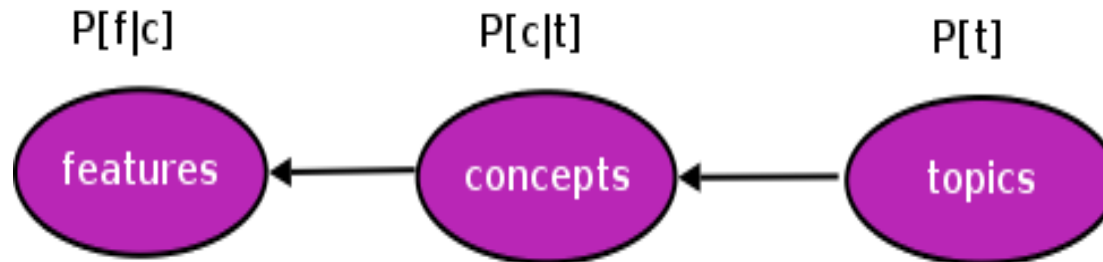
□ Goal

- For a new test document, predict its topic label

- Relate **features** to **topics** through latent **concepts**



□ Generative process for feature-topic pairs



- Select a **topic** t with probability $P[t]$
- Pick a **latent variable** c with probability $P[c|t]$
(Probability that concept c describes topic t)
- Generate a **feature** f with probability $P[f|c]$
(Probability that word f means concept c)

- Associate with each observation (feature f , topic t) a latent variable (concept c)

$$P[f, t] = \sum_{c \in C} P[c] \cdot P[(f, t) | c]$$

- Independence assumptions:
 - Observation pairs (f, t) are generated independently
 - Conditioned on the latent variable c , features f are generated independently of topic t

- Log-likelihood of the observed pairs (f, t) :

$$\log L = \log \left(\prod_{(f, t)} P[f, t]^{n(f, t)} \right)$$

$$l = \sum_{(f, t)} n(f, t) \cdot \log(P[f, t]) = \sum_{(f, t)} n(f, t) \cdot \log \left(\sum_{c \in C} P[c] \cdot P[(f, t) | c] \right)$$

- Estimate model parameters $\{P[t], P[f|c], P[c|t]\}$ so as to maximize the **complete log-likelihood** of (f,t) pairs (EM algorithm):

$$E[l^{comp}] = \sum_t \sum_f n(f,t) \cdot \sum_{c \in C} P[c|(f,t)] \cdot \log(P[t] \cdot P[f|c] \cdot P[c|t])$$

- Use Bayes rule & learned parameters

$$t = \operatorname{argmax}_t P[t|d] = \operatorname{argmax}_t P[d|t] \cdot P[t] = \operatorname{argmax}_t \prod_f P[f,t]$$

$$(P[d|t]) \cdot P[t] = \left(\prod_{f \in d} P[f|t] \right) \cdot P[t] = \prod_{f \in d} P[f,t]$$

$$P[f,t] = \sum_c P[t] \cdot P[f|c] \cdot P[c|t]$$

1. Large number of model parameters

Solution: Prune the parameter space

- ❑ **Feature selection** (Mutual Information)
 - ❑ Extract phrases, exploit PoS information
- ❑ **Concept selection** (Ontology)
 - ❑ Select a subset of concepts from the ontology, that reflects well the semantics of the given training collection
 - ❑ For a given feature f , extract all meanings
 - ❑ Refine this 'mapping' by EM learning

⇒ Reduces computational complexity, Increases model robustness

2. Risk of local maxima

Solution: Pre-initialize model parameters

□ Context based similarity => probability

- Context(f) = text window in document
- Context(c) = hypernyms, hyponyms, siblings + their glosses from ontology
- Context(t) = top features selected by MI from training collection of topic t

$$P[f | c] = \frac{\text{sim}(\text{context}(f), \text{context}(c))}{\sum_{f \in F} \text{sim}(\text{context}(f), \text{context}(c))}, \quad \left(\sum_{f \in F} P[f | c] = 1, \forall c \in C \right)$$

$$P[c | t] = \frac{\text{sim}(\text{context}(c), \text{context}(t))}{\sum_{c \in C} \text{sim}(\text{context}(c), \text{context}(t))}, \quad \left(\sum_{c \in C} P[c | t] = 1, \forall t \in T \right)$$

⇒ Speeds up convergence, Reduces risk of getting stuck in local max

Experimental Results (I)

“Crude oil prices rallied today, moving over 17.00 dls a barrel because of Saudi Arabia's determined effort to support prices, analysts said.”

□ Reuters-21578

- Select top 5 topics: earn, acq, crude, trade, money-fx
- Training: 1,000 documents; Test: 2,000 documents

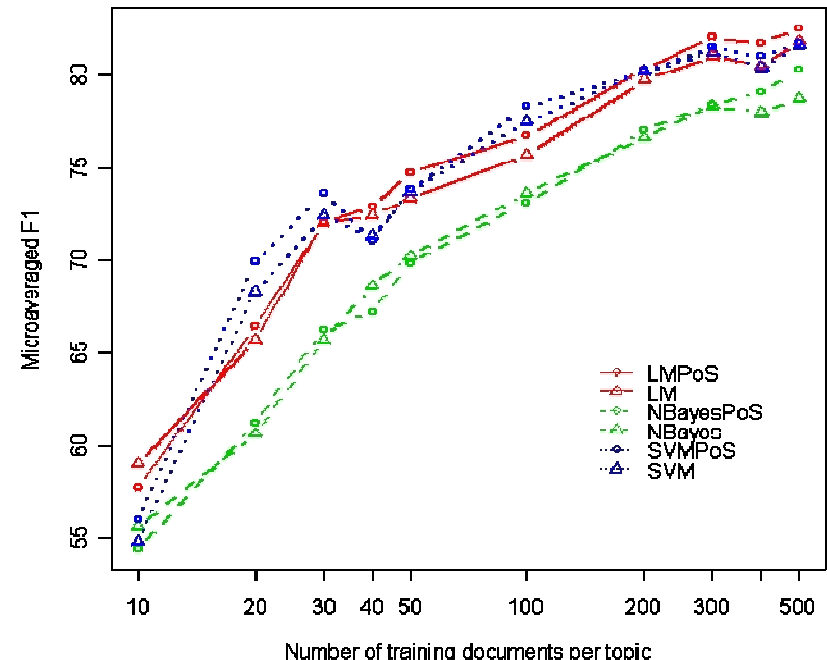
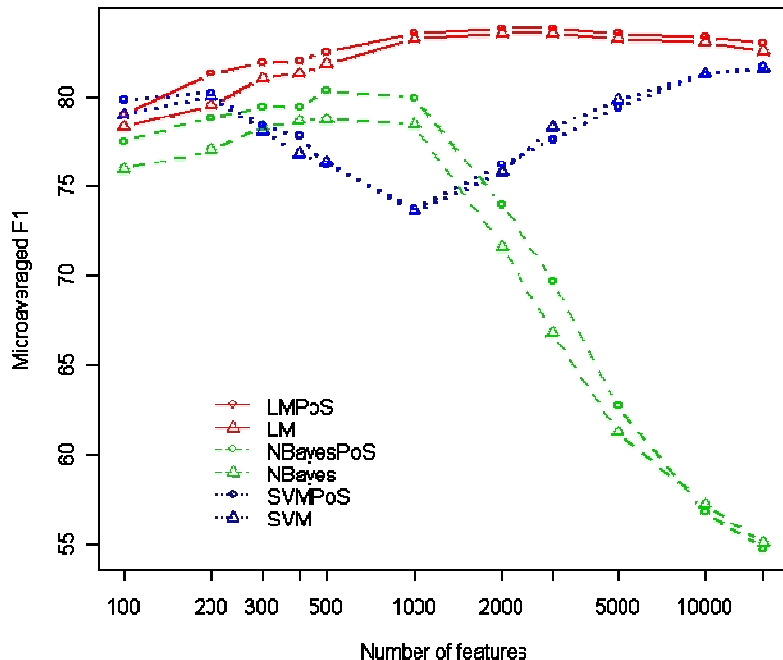
Training per topic	Microavg F1 NBayes	Microavg F1 LatentM	Microavg F1 LatentMPoS	Microavg F1 SVM
10	88.9%	88.7%	87.8%	90.0%
20	89.6%	92.2%	90.7%	92.1%
30	92.7%	94.0%	92.2%	93.6%
40	92.1%	93.0%	91.2%	94.5%
50	93.8%	95.0%	93.8%	93.8%
100	95.3%	94.9%	93.8%	95.5%
150	96.0%	95.0%	94.4%	95.4%
200	95.9%	95.8%	94.5%	95.9%

Experimental Results (II)

In a place where art, science and technology meet, Joseph Scheer's images of moths emerge. These ubiquitous creatures are often considered drab-colored poor relations of the "beautiful" butterfly;

Amazon

- Books' editorial reviews from *amazon.com*
- Select 3 topics: Biological Sciences, Mathematics, Physics
- Training: 1,500 documents; Test: 4,500 documents
- Study **vocabulary size** (left) & **training size influence** (right) on model performance



□ Amazon

- Similarity based heuristic vs. random initialization of parameters
 - Heuristic speeds-up convergence

- Heuristic vs. Heuristic + 1 EM iteration
 - Neither technique alone can achieve good performance

EM ITERATION	SIM-BASED INIT	RANDOM INIT
1	80.5%	59.0%
2	81.5%	70.6%
3	81.9%	76.5%
4	82.2%	79.8%
5	82.3%	80.9%
10	82.5%	82.3%
15	82.5%	82.4%

TRAINING	HEURISTIC	HEURISTIC-EM1	RANDOM-EM1
10	38.1%	56.8%	49.8%
20	66.6%	60.9%	49.6%
30	68.2%	67.7%	49.6%
40	40.3%	70.5%	49.8%
50	43.4%	71.7%	49.8%
100	27.3%	74.8%	49.8%
200	29.9%	79.3%	49.8%
300	27.6%	80.8%	51.0%
400	30.4%	80.3%	51.0%
500	32.3%	80.5%	52.0%

- ❑ Learning word-to-concept mappings can improve accuracy on certain datasets

- ❑ Short-term
 - ❑ Experimental work:
 - ❑ Semantically richer data collections + Customized ontologies (Wikipedia...suggestions welcome)
 - ❑ Different types of classifiers: Bayesian Network
- ❑ Long-term
 - ❑ Given a data collection, predict if & how much semantics can help
 - ❑ Applications of the proposed model in IE, QE

Thank you!