Text Classification

Sentiment Analysis and Opinion Mining

Fabrizio Sebastiani

Qatar Computing Research Institute
Qatar Foundation
PO Box 5825 – Doha, Qatar
E-mail: fsebastiani@qf.org.qa
http://www.qcri.com/

ESSIR 2015, Thessaloniki, GR
31 August – 4 September, 2015

Download most recent version of these slides at
http://bit.ly/1TghbsI

Text Classification

Text classification

1. The task
2. Applications of text classification
3. Supervised learning and text classification
   3.1 Representing text for classification purposes
   3.2 Training a classifier
4. Evaluating a classifier
5. Advanced topics

Text classification

What classification is and is not

- Classification (aka "categorization"): a ubiquitous enabling technology in data science; studied within pattern recognition, statistics, and machine learning.
- Def: the activity of predicting to which among a predefined finite set of groups ("classes", or "categories") a data item belongs to
- Formulated as the task of generating a hypothesis (or "classifier", or "model") $h : \mathcal{D} \rightarrow \mathcal{C}$, where $\mathcal{D} = \{\mathbf{x}_1, \mathbf{x}_2, ...\}$ is a domain of data items and $\mathcal{C} = \{c_1, ..., c_n\}$ is a finite set of classes (the classification scheme)

What classification is and is not (cont'd)

- Different from clustering, where the groups ("clusters") and their number are not known in advance
- In text classification, data items are textual (e.g., news articles, emails, sentences, queries, etc.) or partly textual (e.g., Web pages)
- The membership of a data item into a class must not be determinable with certainty (e.g., predicting whether a natural number belongs to Prime or NonPrime is not classification)

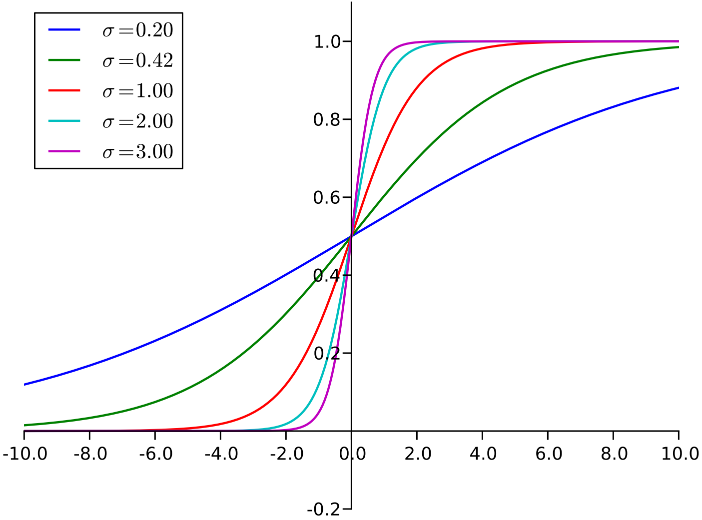
Main types of classification

- Binary classification: $h : \mathcal{D} \rightarrow \mathcal{C}$ (each item belongs to exactly one class) and $\mathcal{C} = \{c_1, c_2\}$
  - E.g., assigning emails to Spam or Legitimate
- Single-Label Multi-Class (SLMC) classification: $h : \mathcal{D} \rightarrow \mathcal{C}$ (each item belongs to exactly one class) and $\mathcal{C} = \{c_1, ..., c_n\}$, with $n > 2$
  - E.g., assigning news articles to one of HomeNews, International, Entertainment, Lifestyles, Sports
- Multi-Label Multi-Class (MLMC) classification: $h : \mathcal{D} \rightarrow 2^{\mathcal{C}}$ (each item may belong to zero, one, or several classes) and $\mathcal{C} = \{c_1, ..., c_n\}$, with $n > 1$
  - E.g., assigning CS articles to classes in the ACM Classification System
  - May be solved as n independent binary classification problems
- Ordinal classification (OC): as in SLMC, but for the fact that $c_1 \preceq ... \preceq c_n$
  - E.g., assigning product reviews to one of Excellent, Good, SoAndSo, Poor, Disastrous

Hard classification and soft classification

- The definitions above denote "hard classification" (HC)
- "Soft classification" (SC) denotes the task of predicting a score for each pair (d, c), where the score denotes the { probability / strength of evidence / confidence } that d belongs to c
  - E.g., a probabilistic classifier outputs "posterior probabilities" $p(c|d) \in [0, 1]$
  - E.g., the AdaBoost classifier outputs scores $s(d, c) \in (-\infty, +\infty)$ that represent its confidence that d belongs to c
  - When scores are not probabilities, they can be converted into probabilities via the use of a sigmoid function

Hard classification and soft classification (cont'd)



Hard classification and soft classification (cont'd)

- Hard classification often consists of
  1. Training a soft classifier that outputs scores $s(d, c)$
  2. Picking a threshold t, such that
     - $s(d, c) > t$ is interpreted as a "Yes"
     - $s(d, c) \leq t$ is interpreted as a "No"
- In soft classification, scores are used for ranking; e.g., ranking items for a given class, ranking classes for a given item
- HC is used for fully autonomous classifiers, while SC is used for interactive classifiers (i.e., with humans in the loop)

Dimensions of classification

- Text classification may be performed according to several dimensions ("axes") orthogonal to each other
- by topic ; by far the most frequent case, its applications are ubiquitous
- by sentiment ; useful in market research, online reputation management, social science and political science
- by language (aka "language identification"); useful, e.g., in query processing within search engines;
- by genre ; e.g., AutomotiveNews vs. AutomotiveBlogs, useful in website classification and others;
- by author (aka "authorship attribution"), by native language ("native language identification"), or by gender ; useful in forensics and cybersecurity
- by usefulness; e.g., product reviews
- ...

Text classification

Example 1. Knowledge organization

- Long tradition in both science and the humanities ; goal was organizing knowledge, i.e., conferring structure to an otherwise unstructured body of knowledge
- Automated classification tries to automate the tedious task of assigning data items based on their content, a task otherwise performed by human annotators (or "assessors")
- The rationale is that using a structured body of knowledge is easier / more effective than if this knowledge is unstructured

Example 1. Knowledge organization (cont'd)

- Scores of applications; e.g.,
  - Classifying news articles for selective dissemination
  - Classifying scientific papers into specialized taxonomies
  - Classifying patents
  - Classifying "classified" ads
  - Classifying answers to open-ended questions
  - Classifying topic-related tweets by sentiment
  - ...
- Retrieval (as in search engines) could also (more in theory than in practice ...) be viewed as (binary + soft) classification into Relevant vs. NonRelevant

Example 2. Filtering

- Filtering (aka "routing") refers to the activity of blocking a set of NonRelevant items from a dynamic stream, thereby leaving only the Relevant ones
  - E.g., when studying the sentiment of Twitter users towards ISIS, tweets that are not about ISIS must be "filtered out"
- Filtering is thus an instance of binary (hard) classification, and its applications are ubiquitous
- Spam filtering is an important example of filtering, attempting to tell Legitimate messages from Spam messages[1]
- Detecting unsuitable content (e.g., porn, violent content, racist content, cyberbullying) also an important application, e.g., in PG filters or on social media

[1] Gordon V. Cormack: Email Spam Filtering: A Systematic Review. *Foundations and Trends in Information Retrieval* 1(4):335–455 (2006)

Example 3. Empowering other IR tasks

- Ancillary to improving the effectiveness of other tasks in IR or NLP; e.g.,
  - Classifying queries by intent within search engines
  - Classifying questions by type in QA systems
  - Classifying named entities
  - Word sense disambiguation in NLP systems
  - ...
- Many of these tasks involve classifying very small texts (e.g., queries, questions, sentences)

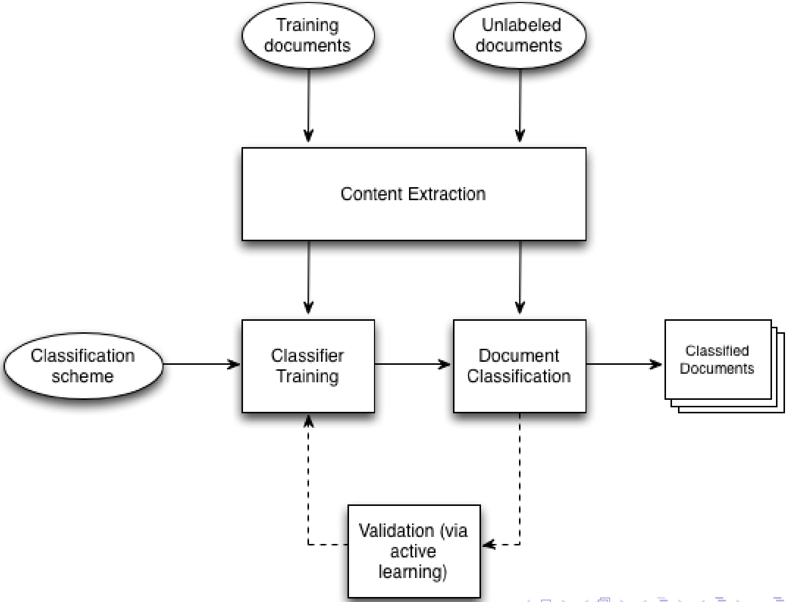
Text classification

1. The task
2. Applications of text classification
3. Supervised learning and text classification
   3.1 Representing text for classification purposes
   3.2 Training a classifier
4. Evaluating a classifier
5. Advanced topics

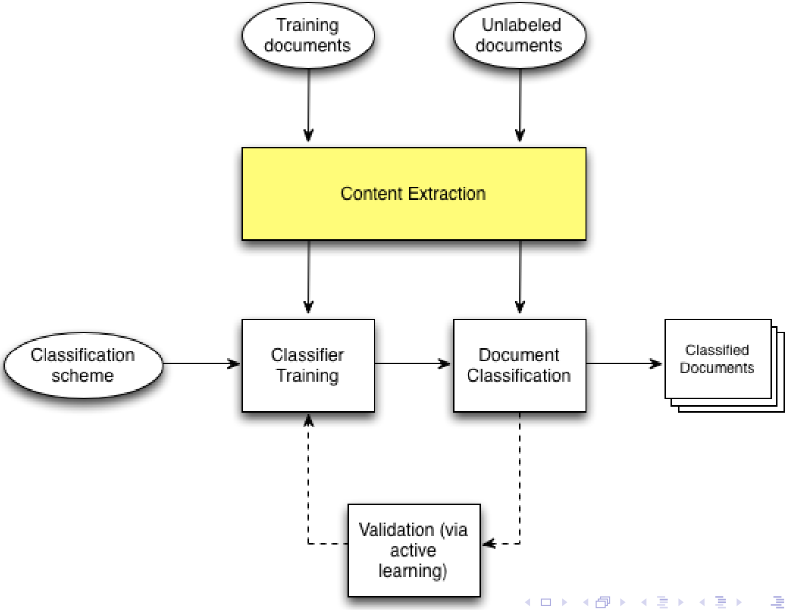
The supervised learning approach to classification

- An old-fashioned way to build text classifiers was via knowledge engineering, i.e., manually building classification rules
  - E.g., (`Viagra` **or** `Sildenafil` **or** `Cialis`) $\rightarrow$ Spam
- Disadvantages: expensive to setup and to mantain
- Superseded by the supervised learning (SL) approach
  - A generic (task-independent) learning algorithm is used to train a classifier from a set of manually classified examples
  - The classifier learns, from these training examples, the characteristics a new text should have in order to be assigned to class c
- Advantages:
  - Generating training examples cheaper than writing classification rules
  - Easy update to changing conditions (e.g., addition of new classes, deletion of existing classes, shifted meaning of existing classes, etc.)

The supervised learning approach to classification



The supervised learning approach to classification



Representing text for classification purposes

- In order to be input to a learning algorithm (or a classifier), all training (or unlabeled) documents are converted into vectors in a common vector space
- The dimensions of the vector space are called features
- In order to generate a vector-based representation for a set of documents D, the following steps need to be taken
  1. Feature Extraction
  2. (Feature Selection *or* Feature Synthesis)
  3. Feature Weighting

Representing text: 1. Feature Extraction

- In classification by topic, a typical choice is to make the set of features coincide with the set of words that occur in D (unigram model, aka “bag-of-words”)
- Word n-grams (i.e., sequences of n words that frequently occur in D – aka “shingles”) may be optionally added; this is usually limited to $n = 2$ (unigram+bigram model)
  - the higher the value of n, the higher the semantic significance and the dimensionality of the resulting representation, and the lower its statistical robustness
- This may be preceded by (a) stop word removal and/or (b) stemming or lemmatization; (b) is meant to improve statistical robustness

Representing text: 1. Feature Extraction (cont'd)

- An alternative to the process above is to make the set of features coincide with the set of character n-grams (e.g., $n \in \{3, 4, 5\}$) that occur in D; useful especially for degraded text (e.g., resulting from OCR or ASR)[2]
- In order to achieve statistical robustness, all of the above renounces to encoding word order and syntactic structure

[2] Paul McNamee, James Mayfield: Character N-Gram Tokenization for European Language Text Retrieval. *Information Retrieval* 7(1-2):73-97 (2004)

Representing text: 1. Feature Extraction

- The above is OK for classification by topic, but not when classifying by other dimensions
- The choice of features for a classification task (feature design) is dictated by the distinctions we want to capture, and is left to the designer; e.g.
  - in classification by author, features such average word length, average sentence length, punctuation frequency, frequency of subjunctive clauses, etc., are used[3]
  - in classification by sentiment, bag-of-words is not enough, and deeper linguistic processing is necessary

[3] Patrick Juola: Authorship Attribution. *Foundations and Trends in Information Retrieval* 1(3): 233-334 (2006)

Representing text: 2a. Feature selection

- Vectors of length $O(10^5)$ or $O(10^6)$ may result, esp. if word n-grams are used; this may give rise to both overfitting and high computational cost;
- Feature selection (FS) has the goal of identifying the most discriminative features, so that the others may be discarded
- The "filter" approach to FS consists in measuring (via a function ϕ) the discriminative power $\nu(t_k)$ of each feature t_k and retaining only the top-scoring features[4]
- For binary classification, a typical choice for ϕ is mutual information, i.e.,

$$IG(t_k|c_i) = \sum_{c \in \{c_i, \overline{c}_i\}} \sum_{t \in \{t_k, \overline{t}_k\}} P(t,c) \log_2 \frac{P(t,c)}{P(t)P(c)}$$

Alternative choices are chi-square and log-odds.

[4] Y. Yang, J. Pedersen: A Comparative Study on Feature Selection in Text Categorization. Proceedings of ICML 1997.

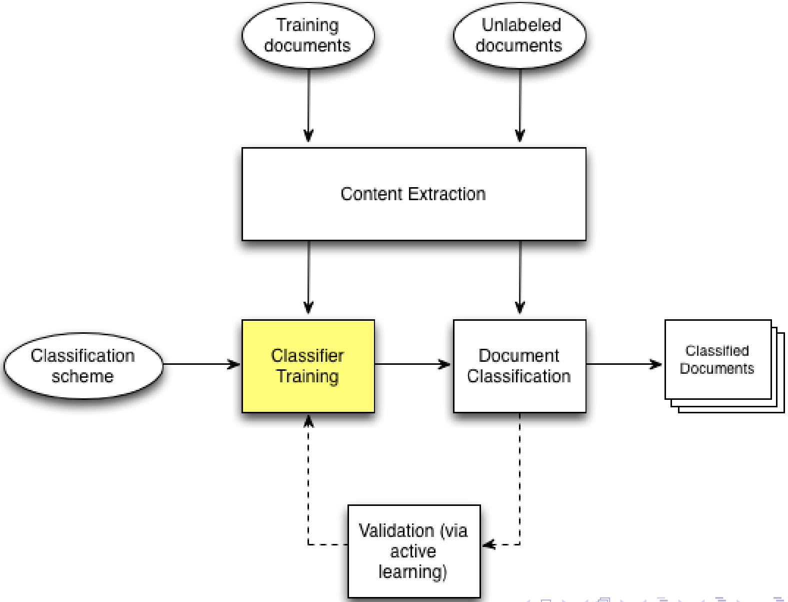
Representing text: 2b. Feature synthesis

- Matrix decomposition techniques (e.g., PCA, SVD, LSA) can be used to synthesize new features that replace the features discussed above
- These techniques are based on the principles of distributional semantics, which states that the semantics of a word "is" the words it co-occurs with in corpora of language use
- The advantage of these techniques is that the synthetic features in the new vectorial representation do not suffer from problems such as polisemy and synonymy
- The disadvantage of these techniques is that they are computationally expensive, sometimes prohibitively so

Representing text: 3. Feature weighting

- Feature weighting means attributing a value to feature t_k in document d_i: this value may be
  - binary (representing presence/absence of t_k in d_i); or
  - numeric (representing the importance of t_k for d_i); obtained via feature weighting functions in the following two classes:
    - unsupervised: e.g, *tfidf* or $BM25$,
    - supervised: e.g., $tf * IG$, $TF * \chi^2$
- The similarity between two vectors may be computed via cosine similarity; if these vectors are pre-normalized, this is equivalent to computing the dot product between them

The supervised learning approach to classification

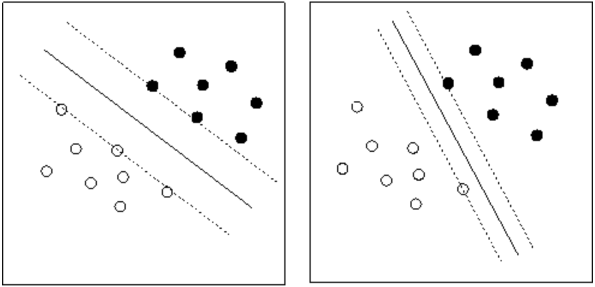


Supervised learning for binary classification

- For binary classification, essentially any supervised learning algorithm can be used for training a classifier; popular choices include
  - Support vector machines (SVMs)
  - Boosted decision stumps
  - Random forests
  - Naïve Bayesian methods
  - Lazy learning methods (e.g., k-NN)
  - ...
- The "No-free-lunch principle" (Wolpert, 1996): $\approx$ there is no learning algorithm that can outperform all others in all contenxts
- Implementations need to cater for
  - the very high dimensionality typical of TC
  - the sparse nature of the representations involved

An example supervised learning method: SVMs

- A constrained optimization problem: find the separating surface (e.g., hyperplane) that maximizes the margin (i.e., the minimum distance between itself and the training examples)



(a) Larger margin (b) Smaller margin

- Margin maximization conducive to good generalization accuracy on unseen data
- Theoretically well-founded + good empirical performance on a variety of tasks
- Publicly available implementations optimized for sparse feature spaces: e.g., SVM-Light, LibSVM, and others

An example supervised learning method: SVMs (cont'd)

- We consider linear separators (i.e., hyperplanes) and classifiers of type

$$h(\mathbf{x}) = sign(\mathbf{w} \cdot \mathbf{x} + b) \tag{1}$$

- Hard-margin SVMs look for

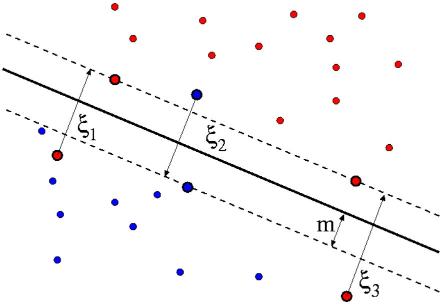
$$\begin{array}{ll} \arg\min_{\mathbf{w} \geq 0} & \frac{1}{2}\mathbf{w} \cdot \mathbf{w} \\ \text{such that} & y_i'[\mathbf{w} \cdot \mathbf{x}_i' + b] \geq 0 \\ & \qquad \text{for all } i \in \{1, ..., |Tr|\} \end{array} \tag{2}$$

- There are now fast algorithms for this[5]

[5] T. Joachims, C.-N. Yu: Sparse kernel SVMs via cutting-plane training. Machine Learning, 2009.

An example supervised learning method: SVMs (cont'd)

- Classification problems are often not linearly separable (LS)

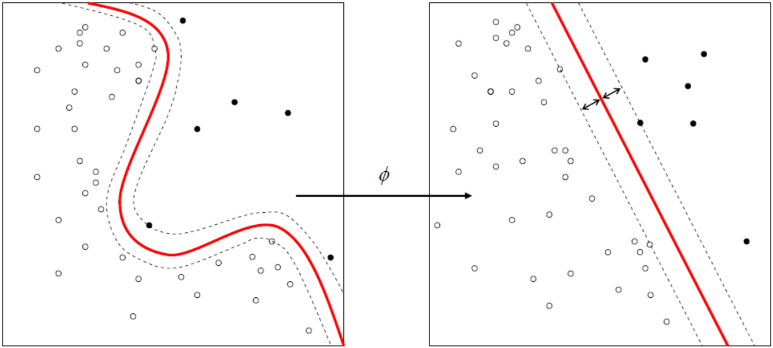


- Soft-margin SVMs introduce penalties for misclassified training examples; they look for

$$
\begin{aligned}
&\arg\min_{\mathbf{w},\xi_i \geq 0} && \frac{1}{2}\mathbf{w}\cdot\mathbf{w} + C\sum_{i=1}^{|Tr|}\xi_i \\
&\text{such that} && y'_i[\mathbf{w}\cdot\mathbf{x}'_i + b] \geq (1-\xi_i) \\
& && \text{for all } i \in \{1, ..., |Tr|\}
\end{aligned}
\tag{3}
$$

An example supervised learning method: SVMs (cont'd)

- Non-LS problems can become LS once mapped to a high-dimensional space



An example supervised learning method: SVMs (cont'd)

- Kernels are similarity functions $K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j)$, where $\phi(\cdot)$ is a mapping into a higher-dimensional space
- SVMs can indeed use kernels instead of the standard dot product; popular kernels are
  - $K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i \cdot \mathbf{x}_j$ (the linear kernel)
  - $K(\mathbf{x}_i, \mathbf{x}_j) = (\gamma \mathbf{x}_i \cdot \mathbf{x}_j + r)^d, \gamma > 0$ (the polynomial kernel)
  - $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma ||\mathbf{x}_i - \mathbf{x}_j||^2), \gamma > 0$ (the RBF kernel)
  - $K(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\gamma \mathbf{x}_i \cdot \mathbf{x}_j + r)$ (the sigmoid kernel)
- However, the linear kernel is usually employed in text classification applications; there are theoretical arguments supporting this[6].

[6] T. Joachims: A Statistical Learning Model of Text Classification for Support Vector Machines. Proceedings of SIGIR 2001.

Supervised learning for non-binary classification

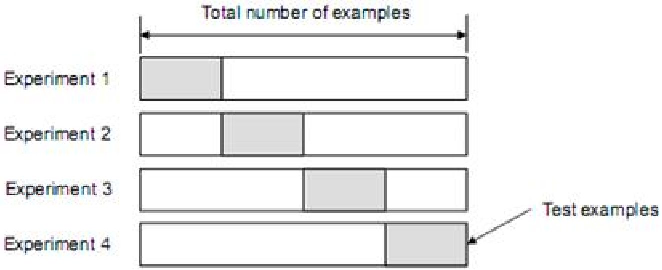
- Some learning algorithms for binary classification are "SLMC-ready"; e.g.
  - Decision trees
  - Boosted decision stumps
  - Random forests
  - Naive Bayesian methods
  - Lazy learning methods (e.g., k-NN)
- For other learners (notably: SVMs) to be used for SLMC classification, combinations / cascades of the binary versions need to be used[7]
- For ordinal classification, algorithms customised to OC need to be used (e.g., SVORIM, SVOREX)[8]

[7] K. Crammer and Y. Singer. On the Algorithmic Implementation of Multi-class SVMs, *Journal of Machine Learning Research*, 2001.

[8] Chu, W., Keerthi, S.: Support vector ordinal regression. *Neural Computation*, 2007.

Parameter optimization in supervised learning

- The trained classifiers often depend on one or more parameters: e.g.,
  - The C parameter in soft-margin SVMs
  - The γ, r, d parameters of non-linear kernels
  - ...
- These parameters need to be optimized, e.g., via k-fold cross-validation



Text classification

Evaluating a classifier

- Two important aspects in the evaluation of a classifier are efficiency and effectiveness
- Efficiency refers to the consumption of computational resources, and has two aspects
  - Training efficiency (also includes time devoted to performing feature selection and parameter optimization)
  - Classification efficiency; usually considered more important than training efficiency, since classifier training is carried out (a) offline and (b) only once
- In text classification papers it is good practice to report training costs and classification costs

Effectiveness

- Effectiveness (aka accuracy) refers to how frequently classification decisions taken by a classifier are "correct"
- Usually considered more important than efficiency, since accuracy issues "are there to stay"
- Effectiveness tests are carried out on one or more datasets meant to simulate operational conditions of use
- The main pillar of effectiveness testing is the evaluation measure we use

Evaluation measures for classification

- Each type of classification (binary/SLMC/MLMC/ordinal) and mode of classification (hard/soft) requires its own measure
- For binary classification, given the contingency table

		true	
		YES	NO
predicted	YES	TP	FP
	NO	FN	TN

  the standard measure is F_1, the harmonic mean of precision (π) and recall (ρ), i.e.,

$$F_1 = \frac{\pi\rho}{\pi+\rho} = \frac{2TP}{2TP+FP+FN}$$

Evaluation measures for classification (cont'd)

- For multi-label multi-class classification, F_1 must be averaged across the classes, according to
  1. microaveraging: compute F_1 from the "collective" contingency table obtained by summing cells (e.g., $TP = \sum_{c_i \in \mathcal{C}} TP_i$)
  2. macroaveraging: compute $F_1(c_i)$ for all $c_i \in \mathcal{C}$ and then average
- Micro usually gives higher scores than macro ...
- For single-label multi-class classification, the most widely used measure is ("vanilla") accuracy, i.e.,

$$A = \frac{TP + TN}{TP + FP + FN + TN}$$

Evaluation measures for classification (cont'd)

- For ordinal classification, the measure must acknowledge that different errors may have different weight; the most widely used one is macroaveraged mean absolute error, i.e.,

$$MAE^M(h, Te) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{|Te_i|}\sum_{x_j \in Te_i}|h(x_j) - y_i| \qquad (4)$$

- For soft classification, measures from the tradiion of ad hoc retrieval are used. E.g., for soft single-label multi-class classification, mean reciprocal ranking can be used, i.e.,

$$MRR(h, Te) = \frac{1}{|Te|}\sum_{x_j \in Te}\frac{1}{r_h(y_i)} \qquad (5)$$

Some datasets for evaluating text classification

	Total examples	Training examples	Test examples	Classes	Hierarchical	Language	Type
Reuters-21578	≈ 13,000	≈ 9,600	≈ 3,200	115	No	EN	MLMC
RCV1-v2	≈ 800,000	≈ 20,000	≈ 780,000	99	Yes	EN	MLMC
20Newsgroups	≈ 20,000	—	—	20	Yes	EN	MLMC
OHSUMED-S	≈ 16,000	≈ 12,500	≈ 3,500	97	Yes	EN	MLMC
TripAdvisor-15763	≈ 15,700	≈ 10,500	≈ 5,200	5	No	EN	Ordinal
Amazon-83713	≈ 83,700	≈ 20,000	≈ 63,700	5	No	EN	Ordinal

Text classification

1. The task
2. Applications of text classification
3. Supervised learning and text classification
   3.1 Representing text for classification purposes
   3.2 Training a classifier
4. Evaluating a classifier
5. Advanced topics

Further topics (sketch)

- Hierarchical classification
  - Classification when the classification scheme has a hierarchical nature
- Hypertext / structured text classification
  - Classification when the items are hypertextual or are structured texts
- Cost-sensitive classification
  - Classification when false positives and false negatives are not equally bad mistakes
- Active learning for classification
  - When the items to label for training purposes are suggested by the system
- Semi-supervised classification
  - When the classifier is trained using a combination of labelled and unlabelled documents

Further topics (cont'd)

- Transductive learning for classification
  - When the classifier is optimized for classifying a specific test set
- Plugging in external knowledge
  - When we want to leverage sources of knowledge other than the training data
- Cross-lingual text classification
  - Learning to classify documents in a language L_t from training data expressed in a language L_s
- Semi-automated text classification
  - Optimizing the work of human assessors that need to check the results
- Text quantification
  - Learning to estimate the distribution of the classes within the unlabelled data

Further reading

- General:
  - C. Aggarwal and C. Zhai: A Survey of Text Classification Algorithms. In C. Aggarwal and C. Zhai (eds.), *Mining Text Data*, pp. 163–222, 2012.
  - T. Joachims: *Learning to Classify Text using Support Vector Machines*. Kluwer, 2002.
- Supervised learning:
  - K. Murphy: *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012.
  - T. Hastie, R. Tibshirani, J. Friedman: *The Elements of Statistical Learning*, 2nd Edition. Springer, 2009.
- Evaluating the effectiveness of text classifiers:
  - N. Japkowicz and M. Shah: *Evaluating Learning Algorithms: A Classification Perspective*. Cambridge University Press, 2011.

Sentiment Analysis and Opinion Mining

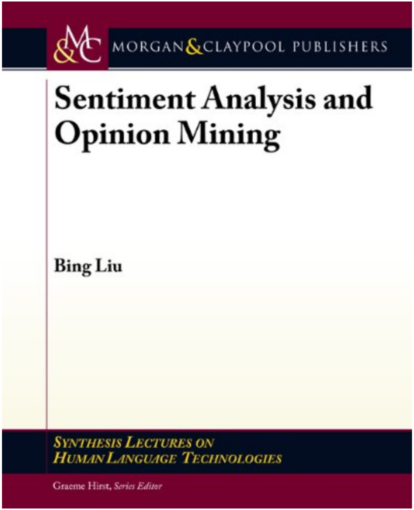
Sentiment Analysis and Opinion Mining

1. The task
2. Applications of SA and OM
3. The main subtasks of SA / OM
4. Advanced topics

Sentiment Analysis and Opinion Mining

1. The task
2. Applications of SA and OM
3. The main subtasks of SA / OM
4. Advanced topics

The task



- Sentiment Analysis and Opinion Mining: a set of tasks concerned with the analysing of texts according to the sentiments / opinions / emotions / judgments (private states, or subjective states) expressed in them
- Originally, term “SA” had a more linguistic slant, while “OM” had a more applicative one
- “SA” and “OM” largely used as synonyms nowadays

Text and "private states"

- SA / OM mostly ignore the difference among what is a "sentiment" / "opinion" / "emotion" / "judgment" ... ; "sentiment" often used to denote all of them

sen•ti•ment |ˈsen(t)əmənt|

noun

1 a view of or attitude toward a situation or event; an opinion : *I agree with your sentiments regarding the road bridge.*

• general feeling or opinion : *the council sought steps to control the rise of racist sentiment.* See note at OPINION .

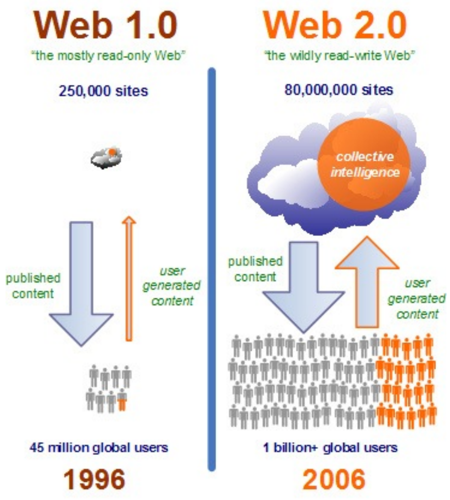
• archaic the expression of a view or desire esp. as formulated for a toast.

2 a feeling or emotion : *an intense sentiment of horror.* See note at EMOTION .

• exaggerated and self-indulgent feelings of tenderness, sadness, or nostalgia : *many of the appeals rely on treacly sentiment.*

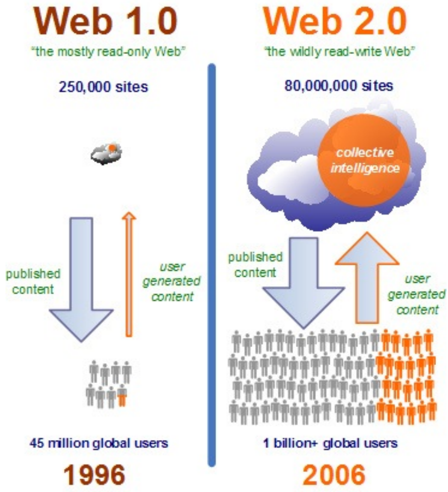
ORIGIN late Middle English (in the senses [personal experience] and [physical feeling, sensation]): from Old French ***sentement***, from medieval Latin ***sentimentum***, from Latin ***sentire*** ***'feel.'***

Opinion mining and the "Web 2.0"



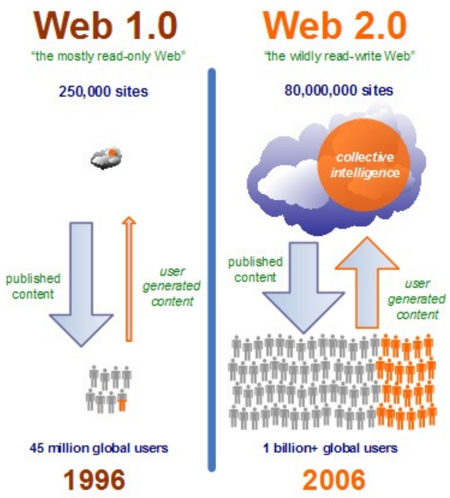
- Opinion mining "is born" with Web 2.0.
- The '90s:
  - Web 1.0 mostly a repository of factual or functional content; authors of content are mainly professional users; non-professional users are mostly "passive" users.
  - SA still an esoteric academic exercise in NLP

Opinion mining and the Web 2.0 (cont.)



- The 2000's: Web 2.0 is born
- The following gradually become popular:
  - blogs and blogging services
  - social networking services (Facebook, Twitter, YouTube, etc.)
  - services that, among other things, host "user-generated content" (Amazon, Yelp, TripAdvisor, Epinions, etc.)
- Non-professional users also become authors of content, and this content is often opinion-laden.

Opinion mining and the Web 2.0 (cont.)



- With the growth of UGC, companies understand the value of these data (e.g., product reviews), and generate the demand for technologies capable of mining "sentiment" from them.
- SA becomes the "Holy Grail" of market research, opinion research, and online reputation management.
- For the 1st time in history we have a huge volume of opinion-rich data ready for analysis in digital form.

Sentiment Analysis and Opinion Mining

1. The task
2. Applications of SA and OM
3. The main subtasks of SA / OM
4. Advanced topics

Opinion research / market research via surveys

- Questionnaires may contain "open" questions
- In many such cases the opinion dimension needs to be analysed, esp. in
  - social sciences surveys
  - political surveys
  - customer satisfaction surveys
- Many such applications are instances of mixed topic / sentiment classification

Computational Social Science

plos.org create account sign in

PLOS | ONE Subject Areas For Authors About Us Search

advanced search

OPEN ACCESS PEER-REVIEWED

30,985	19	131	97
VIEWS	CITATIONS	SAVES	SHARES

RESEARCH ARTICLE

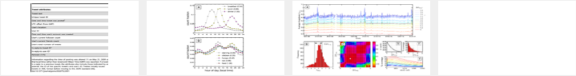
Temporal Patterns of Happiness and Information in a Global Social Network: Hedonometrics and Twitter

Peter Sheridan Dodds, Kameron Decker Harris, Isabel M. Kloumann, Catherine A. Bliss, Christopher M. Danforth

Article | About the Authors | Metrics | Comments | Related Content

Download PDF

Print Share



Hide Figures

Subject Areas

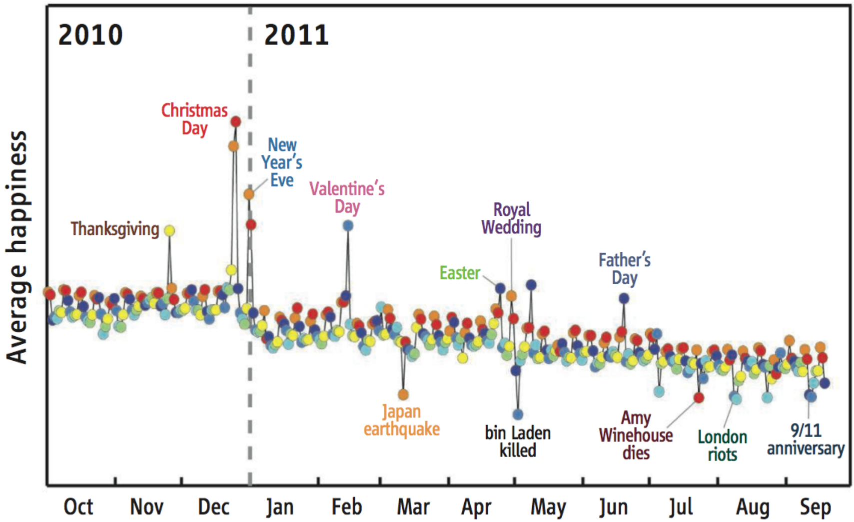
- Ecological metrics
- Psychometrics
- Entropy
- Social sciences
- Earthquakes
- Behavior
- Language
- Graphs

- Abstract
- Introduction
- Results and Discussion
- Methods
- Supporting Information

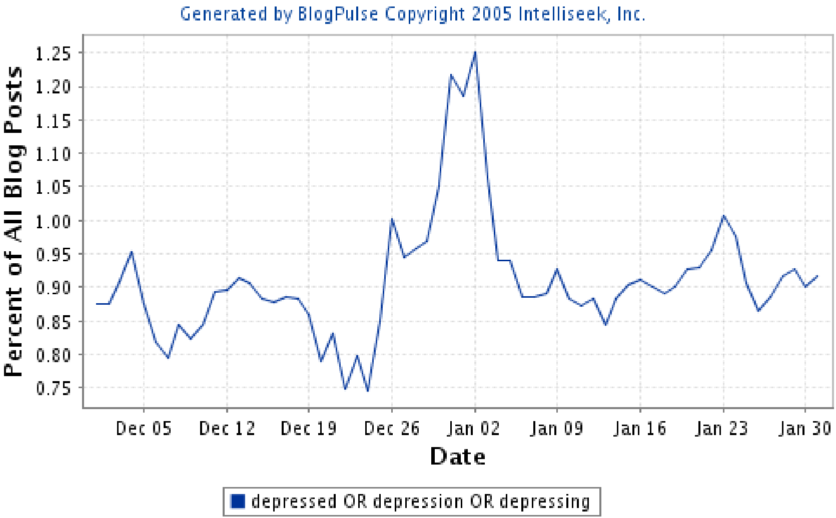
Abstract

Individual happiness is a fundamental societal metric. Normally measured through self-report, happiness has often been indirectly characterized and overshadowed by more readily quantifiable economic indicators such as gross domestic product. Here, we examine expressions made on the online, global microblog and social networking service Twitter,

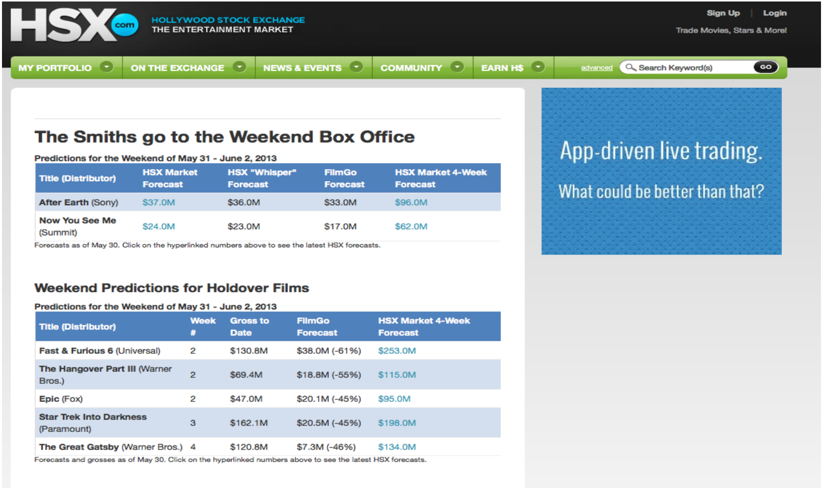
Computational Social Science (cont.)



Computational Social Science (cont.)



Market Research via Social Media Analysis



The Smiths go to the Weekend Box Office

Predictions for the Weekend of May 31 - June 2, 2013

Title (Distributor)	HSX Market Forecast	HSX "Whisper" Forecast	FilmGo Forecast	HSX Market 4-Week Forecast
After Earth (Sony)	$37.0M	$36.0M	$33.0M	$96.0M
Now You See Me (Summit)	$24.0M	$23.0M	$17.0M	$62.0M

Forecasts as of May 30. Click on the hyperlinked numbers above to see the latest HSX forecasts.

Weekend Predictions for Holdover Films

Predictions for the Weekend of May 31 - June 2, 2013

Title (Distributor)	Week #	Gross to Date	FilmGo Forecast	HSX Market 4-Week Forecast
Fast & Furious 6 (Universal)	2	$130.8M	$38.0M (-61%)	$253.0M
The Hangover Part III (Warner Bros.)	2	$69.4M	$18.8M (-55%)	$115.0M
Epic (Fox)	2	$47.0M	$20.1M (-45%)	$95.0M
Star Trek Into Darkness (Paramount)	3	$162.1M	$20.5M (-45%)	$198.0M
The Great Gatsby (Warner Bros.)	4	$120.8M	$7.3M (-46%)	$134.0M

Forecasts and grosses as of May 30. Click on the hyperlinked numbers above to see the latest HSX forecasts.

Market Research via Social Media Analysis (cont.)

DSpace@MIT

Search (Ex: crystalline silicon solar)

Search Within This Collection   Advanced Search

Home → MIT Libraries → MIT Theses → Theses - Dept. of Electrical Engineering and Computer Sciences → Electrical Engineering and Computer Sciences - Master's degree → View Item

▾ Browse

All of DSpace@MIT
- Communities & Collections
- By Issue Date
- Authors
- Titles
- Subjects

This Collection
- By Issue Date
- Authors
- Titles
- Subjects

▸ My Account

Using sentiment and social network analyses to predict opening-movie box-office success

Download

Author: Doshi, Lyric (Lyric Pankaj)

Citable URI: http://hdl.handle.net/1721.1/61284

Other Contributors: Massachusetts Institute of Technology. Dept. of Electrical Engineering and Computer Science.

Advisor: Peter Gloor.

Department: Massachusetts Institute of Technology. Dept. of Electrical Engineering and Computer Science.

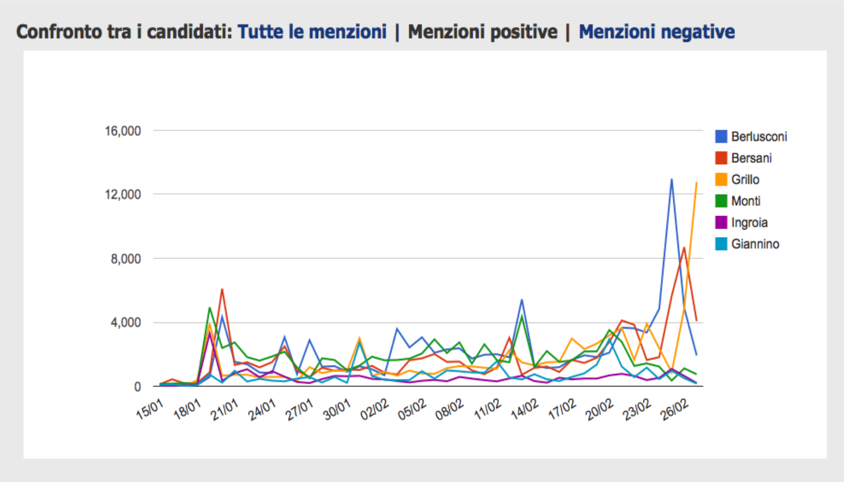
Publisher: Massachusetts Institute of Technology

Date Issued: 2010

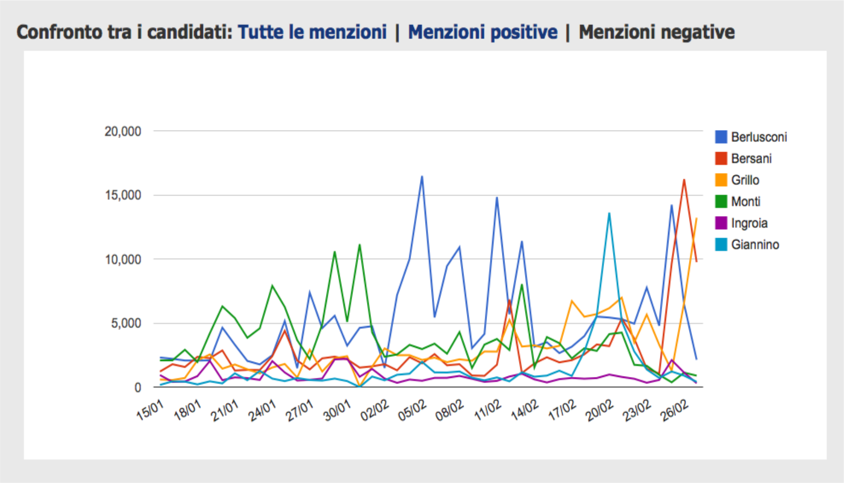
Abstract:

In this thesis, we explore notions of collective intelligence in the form of web metrics, social network analysis and sentiment analysis to predict the box-office income of movies. Successful prediction

Political Science: Predicting election results (cont.)

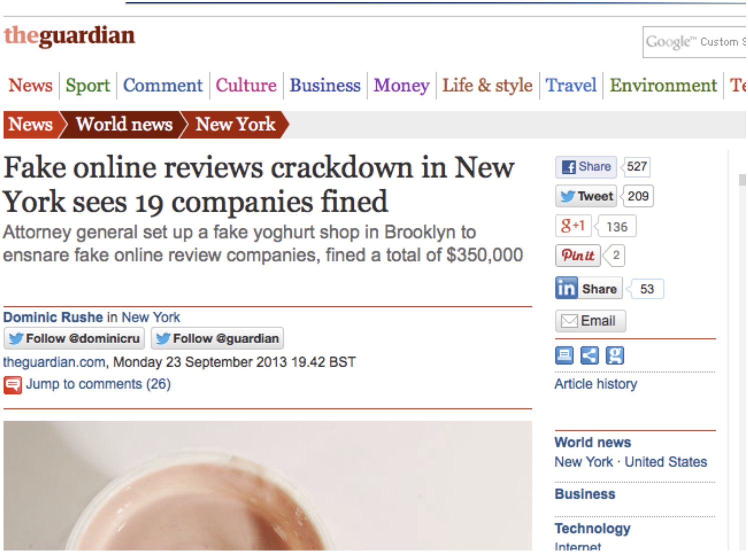


Political Science: Predicting election results (cont.)



Online reputation detection / management

Online reputation detection / management (cont.)



theguardian

Google™ Custom S

News | Sport | Comment | Culture | Business | Money | Life & style | Travel | Environment | Te

News › World news › New York

Fake online reviews crackdown in New York sees 19 companies fined

Attorney general set up a fake yoghurt shop in Brooklyn to ensnare fake online review companies, fined a total of $350,000

Dominic Rushe in New York

Follow @dominicru Follow @guardian

theguardian.com, Monday 23 September 2013 19.42 BST

Jump to comments (26)

Share 527

Tweet 209

+1 136

Pin it 2

Share 53

Email

Article history

World news

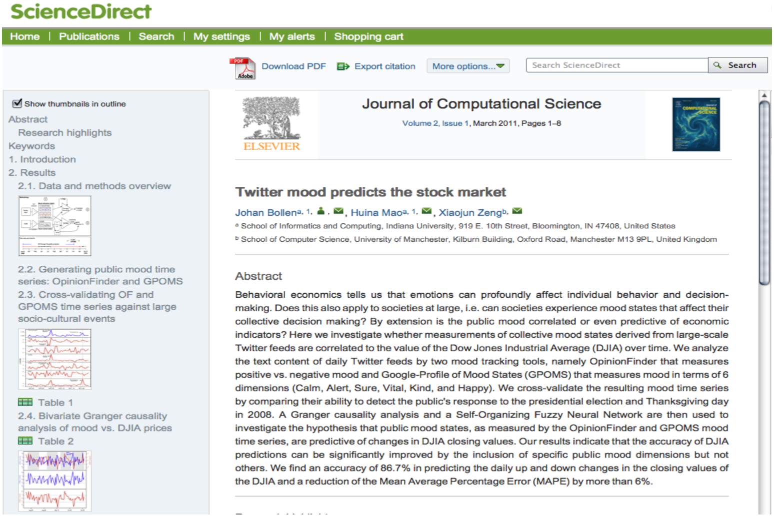
New York · United States

Business

Technology

Internet

Stock change prediction


ScienceDirect

Home | Publications | Search | My settings | My alerts | Shopping cart

Download PDF Export citation More options... Search ScienceDirect Search

Show thumbnails in outline

Abstract

Research highlights

Keywords

1. Introduction

2. Results

2.1. Data and methods overview

2.2. Generating public mood time series: OpinionFinder and GPOMS

2.3. Cross-validating OF and GPOMS time series against large socio-cultural events

Table 1

2.4. Bivariate Granger causality analysis of mood vs. DJIA prices

Table 2

ELSEVIER

Journal of Computational Science

Volume 2, Issue 1, March 2011, Pages 1–8

Twitter mood predicts the stock market

Johan Bollen[a, 1], Huina Mao[a, 1], Xiaojun Zeng[b]

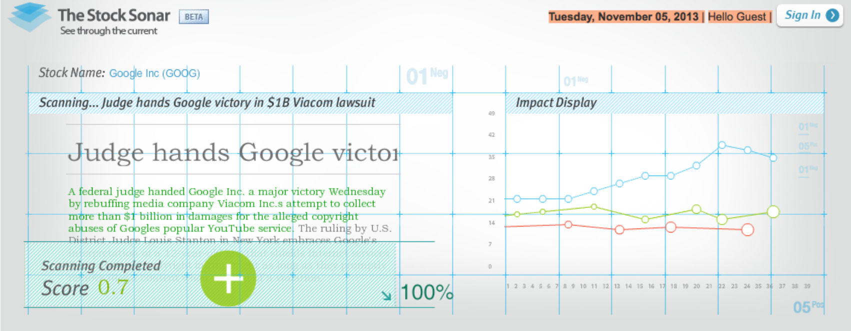
[a] School of Informatics and Computing, Indiana University, 919 E. 10th Street, Bloomington, IN 47408, United States

[b] School of Computer Science, University of Manchester, Kilburn Building, Oxford Road, Manchester M13 9PL, United Kingdom

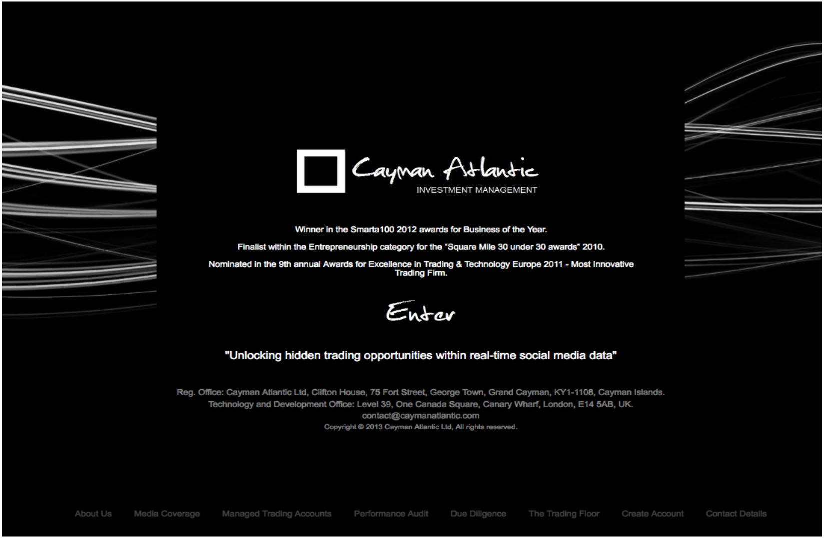
Abstract

Behavioral economics tells us that emotions can profoundly affect individual behavior and decision-making. Does this also apply to societies at large, i.e. can societies experience mood states that affect their collective decision making? By extension is the public mood correlated or even predictive of economic indicators? Here we investigate whether measurements of collective mood states derived from large-scale Twitter feeds are correlated to the value of the Dow Jones Industrial Average (DJIA) over time. We analyze the text content of daily Twitter feeds by two mood tracking tools, namely OpinionFinder that measures positive vs. negative mood and Google-Profile of Mood States (GPOMS) that measures mood in terms of 6 dimensions (Calm, Alert, Sure, Vital, Kind, and Happy). We cross-validate the resulting mood time series by comparing their ability to detect the public's response to the presidential election and Thanksgiving day in 2008. A Granger causality analysis and a Self-Organizing Fuzzy Neural Network are then used to investigate the hypothesis that public mood states, as measured by the OpinionFinder and GPOMS mood time series, are predictive of changes in DJIA closing values. Our results indicate that the accuracy of DJIA predictions can be significantly improved by the inclusion of specific public mood dimensions but not others. We find an accuracy of 86.7% in predicting the daily up and down changes in the closing values of the DJIA and a reduction of the Mean Average Percentage Error (MAPE) by more than 6%.

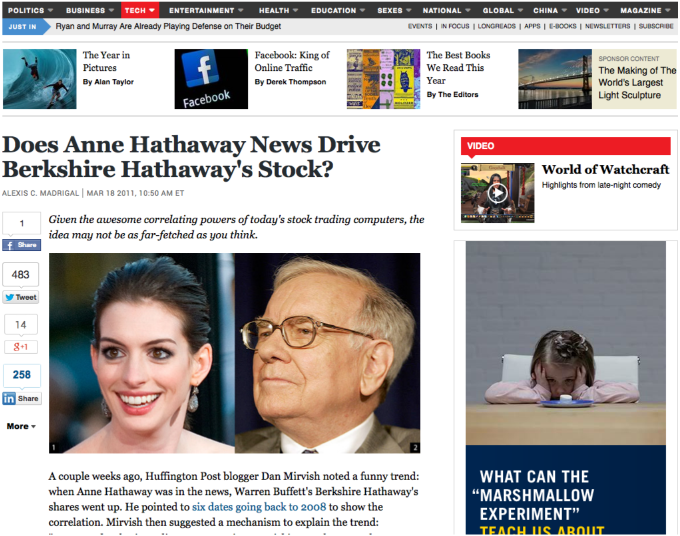
Stock change prediction (cont.)



Stock change prediction (cont.)



Stock change prediction (cont.)



POLITICS | BUSINESS | TECH | ENTERTAINMENT | HEALTH | EDUCATION | SEXES | NATIONAL | GLOBAL | CHINA | VIDEO | MAGAZINE

JUST IN Ryan and Murray Are Already Playing Defense on Their Budget

EVENTS | IN FOCUS | LONGREADS | APPS | E-BOOKS | NEWSLETTERS | SUBSCRIBE

The Year in Pictures
By Alan Taylor

Facebook: King of Online Traffic
By Derek Thompson

The Best Books We Read This Year
By The Editors

SPONSOR CONTENT
The Making of The World's Largest Light Sculpture

Does Anne Hathaway News Drive Berkshire Hathaway's Stock?

ALEXIS C. MADRIGAL | MAR 18 2011, 10:50 AM ET

1 Share

483 Tweet

14 +1

258 Share

More

Given the awesome correlating powers of today's stock trading computers, the idea may not be as far-fetched as you think.

A couple weeks ago, Huffington Post blogger Dan Mirvish noted a funny trend: when Anne Hathaway was in the news, Warren Buffett's Berkshire Hathaway's shares went up. He pointed to six dates going back to 2008 to show the correlation. Mirvish then suggested a mechanism to explain the trend:

VIDEO

World of Watchcraft

Highlights from late-night comedy

WHAT CAN THE "MARSHMALLOW EXPERIMENT" TEACH US ABOUT

Computational advertising

Computational advertising (cont.)

11-year-old charged with driving drunk

REUTERS

NEWS ALERTS

Get an alert when there are new stories about:

- Orange Beach, Alabama
- Perdido Key, Florida
- Chevrolet Monte Carlo

Add Selected Alerts

» More Alerts

ELSEWHERE ON THE WEB

CNN.COM
Explosion rocks besieged mosque in Pakistani capital

ABC NEWS
'Out of the Blue': Do Aliens Exist?

THE CHRISTIAN SCIENCE MONITOR
William Bratton: Lauded chief of troubled LAPD

Beam Me Up

SCI-FI CONVENTION

Underground

Fri Jul 6, 3:23 PM ET

MIAMI (Reuters) - An 11-year-old girl was charged with drunken driving after leading police on a chase at speeds of up to 100 mph that ended when she flipped the car in an Alabama beach town.

A video camera in the police car captured the look of surprise on the officer's face when he approached the wrecked car and got a look at the motorist.

ADVERTISEMENT

HOORay BeeR!

Click here for more BOO

Red Stripe

LAGER

The Mobile Press-Register newspaper said the patrolman saw the Chevrolet Monte Carlo speeding and flashed his lights to signal the driver to stop. Instead, the car sped faster, traveling at up to 100 mph (160 kph) before sideswiping another vehicle and flipping over in the Gulf Coast town of Orange Beach, Alabama, on Tuesday night.

The young driver, who lived nearby in Perdido Key, Florida, was treated at a hospital for scrapes and bruises and released to relatives. Police also charged her with speeding, leaving the scene of an accident and reckless endangerment.

The car belonged to a relative and police were still trying to find out where she got the alcohol. There was none in the vehicle but her blood alcohol level was over the limit for adult motorists, police told the newspaper.

Email Story · IM Story · Printable View

Sentiment Analysis and Opinion Mining

1. The task
2. Applications of SA and OM
3. The main subtasks of SA / OM
4. Advanced topics

A few milestones in SA / OM

- 1997: 1st attempt at automatically generating a sentiment lexicon
  - V. Hatzivassiloglou, K. McKeown: Predicting the Semantic Orientation of Adjectives. ACL 1997
- 2002: 1st works on (binary) "sentiment classification"
  - P. Turney: Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. ACL 2002
  - B. Pang, L. Lee, S. Vaithyanathan: Thumbs up? Sentiment Classification using Machine Learning Techniques. CoRR 2002
- 2004– : SA goes mainstream;
- 2005: 1st work on "ordinal classification by sentiment"
  - B. Pang, L. Lee: Seeing Stars: Exploiting Class Relationships for Sentiment Categorization with Respect to Rating Scales. ACL 2005
- 2006: 1st large-coverage sentiment lexicon
  - A. Esuli, F. Sebastiani. SentiWordNet: A Publicly Available Lexical Resource for Opinion Mining. LREC 2006
- 2008: 1st investigation of aspect-based sentiment
  - I. Titov, R. McDonald: A Joint Model of Text and Aspect Ratings for Sentiment Summarization. ACL 2008

How difficult is sentiment analysis?

- Sentiment analysis is inherently difficult, because in order to express opinions / emotions / etc. we often use a wide variety of sophisticated expressive means (e.g., metaphor, irony, sarcasm, allegation, understatement, etc.)
  - "At that time, Clint Eastwood had only two facial expressions: with the hat and without it."
    (from an interview with Sergio Leone)
  - "She runs the gamut of emotions from A to B"
    (on Katharine Hepburn in "The Lake", 1934)
  - "If you are reading this because it is your darling fragrance, please wear it at home exclusively, and tape the windows shut."
    (from a 2008 review of parfum "Amarige", Givenchy)
- Sentiment analysis could be characterised as an "NLP-complete" problem

Main subtasks within SA / OM

- Sentiment Classification: classify a piece of text based on whether it expresses a Positive / Neutral / Negative sentiment
- Sentiment Lexicon Generation: determine whether a word / multiword conveys a Positive, Neutral, or Negative stance
- Sentiment Quantification: given a set of texts, estimate the prevalence of different Positive, Neutral, Negative sentiments
- Opinion Extraction (aka "Fine-Grained SA"): given an opinion-laden sentence, identify the holder of the opinion, its object, its polarity, the strength of this polarity, the type of opinion
- Aspect-Based Sentiment Extraction: given an opinion-laden text about an object, estimate the sentiments conveyed by the text concerning different aspects of the object

Sentiment Classification

- The "queen" of OM tasks
- May be topic-biased or not
  1. Classify items by sentiment; vs.
  2. Find items that express an opinion about the topic, and classify them by their sentiment towards the topic
- Binary, ternary, or n-ary (ordinal) versions
  - Ternary also involves Neutral or Lukewarm (sometimes confusing the two ...)
  - Ordinal typically uses 1stars, 2Stars, 3Stars, 4Stars, 5Stars as classes
- At the sentence, paragraph, or document level
  - Classification at the more granular levels used to aid classification at the less granular ones
- May be supervised or unsupervised

Sentiment Classification (cont'd)

- Unsupervised Sentiment Classification (USC) relies on a sentiment lexicon
- The first USC approaches just the number of occurrences of Positive words and Negative words in the text
- Approach later refined in various ways; e.g.,
  - If topic-biased, measure the distance between the sentiment-laden word and a word denoting the topic
  - Bring to bear valence shifters (e.g., particles indicating negated contexts such as `not`, `hardly`, etc.)
  - Bring to bear magnifiers (e.g., `very`, `extremely`) and diminishers (e.g., `fairly`)
  - Bring in syntactic analysis (and other levels of linguistic processing) to determine if sentiment *really* applies to the topic
  - Use WSD in order to better exploit sense-level sentiment lexicons

Sentiment Classification (cont'd)

- Supervised Sentiment Classification (SSC) is just (single-label) text classification with sentiment-related polarities as the classes
- Key fact: bag-of-words does not lead anywhere ...
  - E.g., "`A horrible hotel in a beautiful town!`" vs. "`A beautiful hotel in a horrible town!`"
- The same type of linguistic processing used for USC is also needed for SC, with the goal of generating features for vectorial representations
- Supervised tends to work better, but requires training data; this has spawned research into
  - Semi-supervised sentiment classification
  - Transfer learning for sentiment classification

Sentiment Lexicon Generation

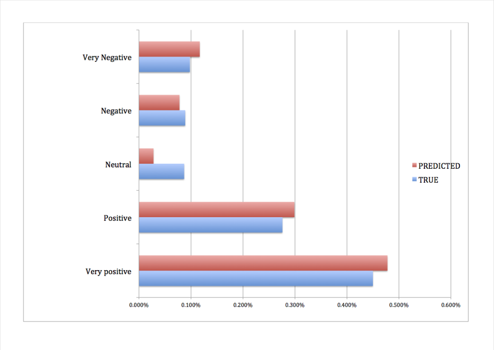
- The use of a sentiment lexicon is central to both USC and SSC (and to all other OM-related tasks)
- Early sentiment lexicons were small, at the word level, and manually annotated
  - E.g., the General Inquirer
- SLs generated from corpora later become dominant;
  - Some of them are at the word sense level (e.g., SentiWordNet)
  - Some of them are medium-dependent (e.g., SLs for Twitter)
  - Some of them are domain-dependent (e.g., SLs for the financial domain)
  - Many of them are for languages other than English (e.g., SentiWordNet's in other languages)

Sentiment Lexicon Generation (cont'd)

- Several intuitions can be used to generate / extend a SL automatically; e.g.,
  - Conjunctions tend to indicate similar polarity ("`cozy and comfortable`") or opposite polarity ("`small but cozy`") (Hatzivassiloglou and McKeown, 1997)
  - Adjectives highly correlated to adjectives with known polarity tend to have the same polarity (Turney and Littman, 2003)
  - Synonyms (indicated as such in standard thesauri) tend to have the same polarity, while antonyms tend to have opposite polarity (Kim and Hovy, 2004)
  - Sentiment classification of words may be accomplished by classifying their definitions (Esuli and Sebastiani, 2005)
  - Words used in definitions tend to have the same polarity as the word being defined (Esuli and Sebastiani, 2007)
- The main problem related to SLs is that the polarity of words / word senses is often context-dependent (e.g., `warm blanket` vs. `warm beer`; `low interest rates` vs. `low ROI`)

Sentiment Quantification

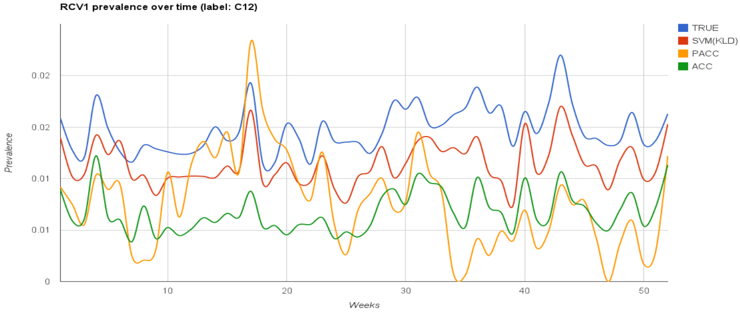
- In many applications of sentiment classification (e.g., market research, social sciences, political sciences), estimating the relative proportions of Positive / Neutral / Negative documents is the real goal; this is called sentiment quantification[9]
  - E.g., tweets, product reviews



[9] A. Esuli and F. Sebastiani. Sentiment Quantification. *IEEE Intelligent Systems*, 2010.

Sentiment quantification (cont'd)

- Tackling quantification via classification is suboptimal, since classifiers (even good ones) may be biased



- Algorithms specific to quantification (e.g., SVM(KLD)) deliver better quantification accuracy[10]

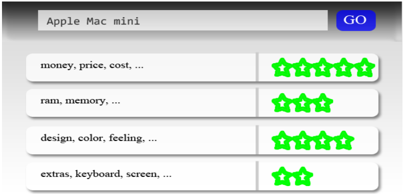
[10] W. Gao and F. Sebastiani. Tweet Sentiment: From Classification to Quantification. Proceedings of ASONAM 2015.

Other tasks

- Opinion Extraction (aka "Fine-Grained SA"): given an opinion-laden sentence, identify the holder of the opinion, its object, its polarity, the strength of this polarity, the type of opinion
  - An instance of information extraction, usually carried out via sequence learning
  - More difficult that standard IE; certain concepts may be instantiated only implicitly

Other tasks (cont'd)

- Aspect-Based Sentiment Extraction: given an opinion-laden text about an object, estimate the sentiments conveyed by the text concerning different aspects of the object
  - Largely driven by need of mining / summarizing product reviews



  - Heavily based on extracting NPs (e.g., `wide viewing angle`) that are highly correlated with the product category (e.g., `Tablet`).
  - Aspects (e.g., `angle`) and sentiments (e.g., `wide viewing`) can be robustly identified via mutual reinforcement

Shared tasks related to sentiment analysis

- Task 4: Sentiment Analysis in Twitter
  - Subtask A: Tweet Polarity Classification
  - Subtask B: Topic-Based Tweet Polarity Classification:
  - Subtask C: Tweet classification according to a five-point scale
  - Subtask D: Tweet quantification according to a two-point scale
  - Subtask E: Tweet quantification according to a five-point scale
- Task 5: Aspect-Based Sentiment Analysis
- Task 6: Detecting Stance in Tweets
- Task 7: Determining Sentiment Intensity of English and Arabic Phrases

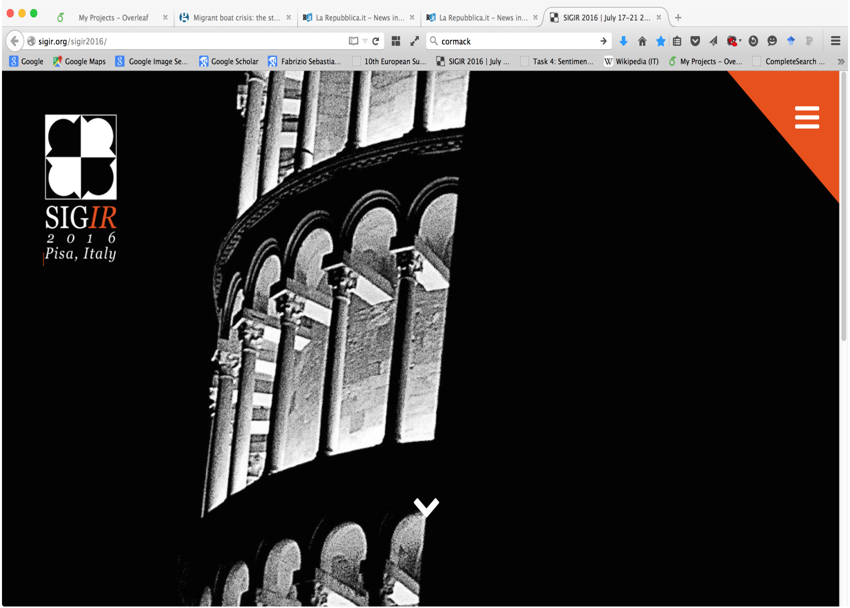
Advanced topics in sentiment analysis

- Automatic generation of context-sensitive lexicons
- Lexemes as complex objects in sentiment lexicons
- Making sense of sarcasm / irony
- Detecting emotion / sentiment in audio / video using non-verbal features
- Cross-domain sentiment analysis
- Cross-lingual / cross-cultural sentiment analysis

Further reading

- General:
  - B. Pang, L. Lee: Opinion Mining and Sentiment Analysis. *Foundations and Trends in Information Retrieval*, 2007.
  - B. Liu: *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers, 2012.
  - R. Feldman: Techniques and applications for sentiment analysis. *Communications of the ACM*, 2013.
- Sentiment analysis in social media
  - S. Kiritchenko, X. Zhu, S. Mohammad: Sentiment Analysis of Short Informal Texts. *Journal of Artificial Intelligence Research*, 2014.
  - Martínez-Càmara, E., Martín-Valdivia, M., Urenã López, L., and Montejo Ráez, A. (2012). Sentiment analysis in Twitter. *Natural Language Engineering*, 2014.

Questions?


My Projects – Overleaf
Migrant boat crisis: the st...
La Repubblica.it – News in...
La Repubblica.it – News in...
SIGIR 2016 | July 17–21 2...
sigir.org/sigir2016/
cormack
Google
Google Maps
Google Image Se...
Google Scholar
Fabrizio Sebastia...
10th European Su...
SIGIR 2016 | July ...
Task 4: Sentimen...
Wikipedia (IT)
My Projects – Ove...
CompleteSearch ...
SIGIR
2 0 1 6
Pisa, Italy

Thank you!

For any question, email me at
fsebastiani@qf.org.qa