Machine Learning for Language Processing
ACS 2015/16
Stephen Clark
L1: Classification

UNIVERSITY OF CAMBRIDGE

The ML Revolution in NLP

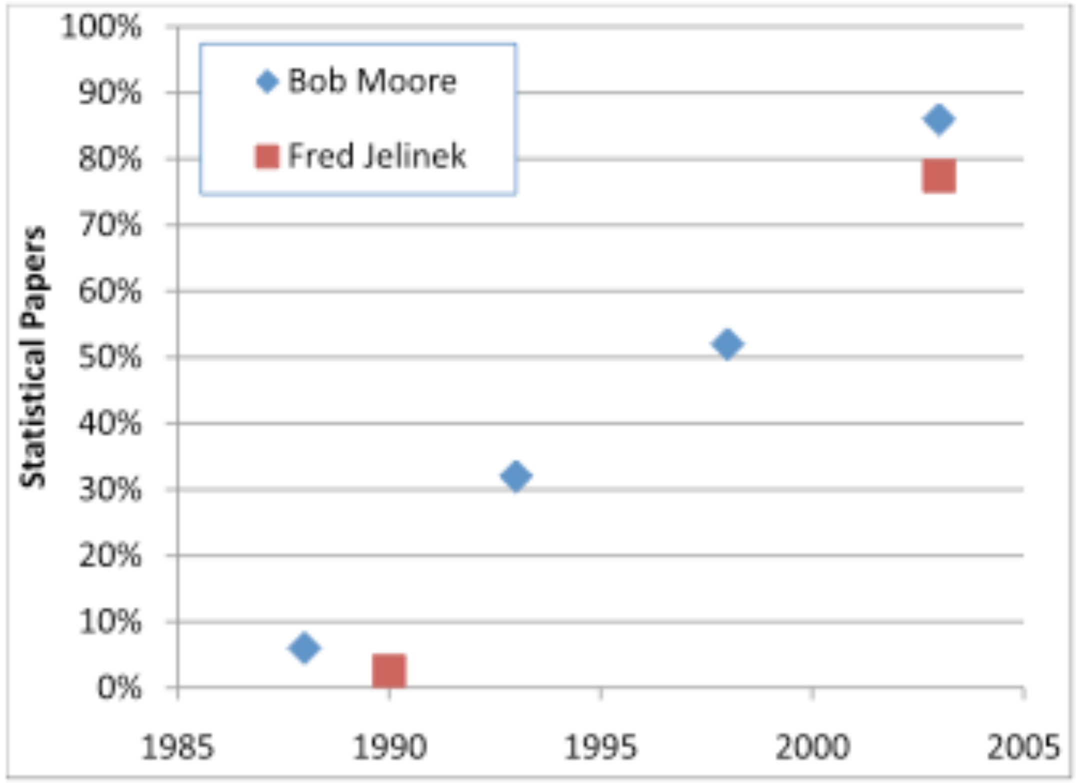


FIGURE 1 The shift from Rationalism to Empiricism is striking (and no longer controversial). This plot is based on two independent surveys of ACL meetings by Bob Moore and Fred Jelinek (personal communication).

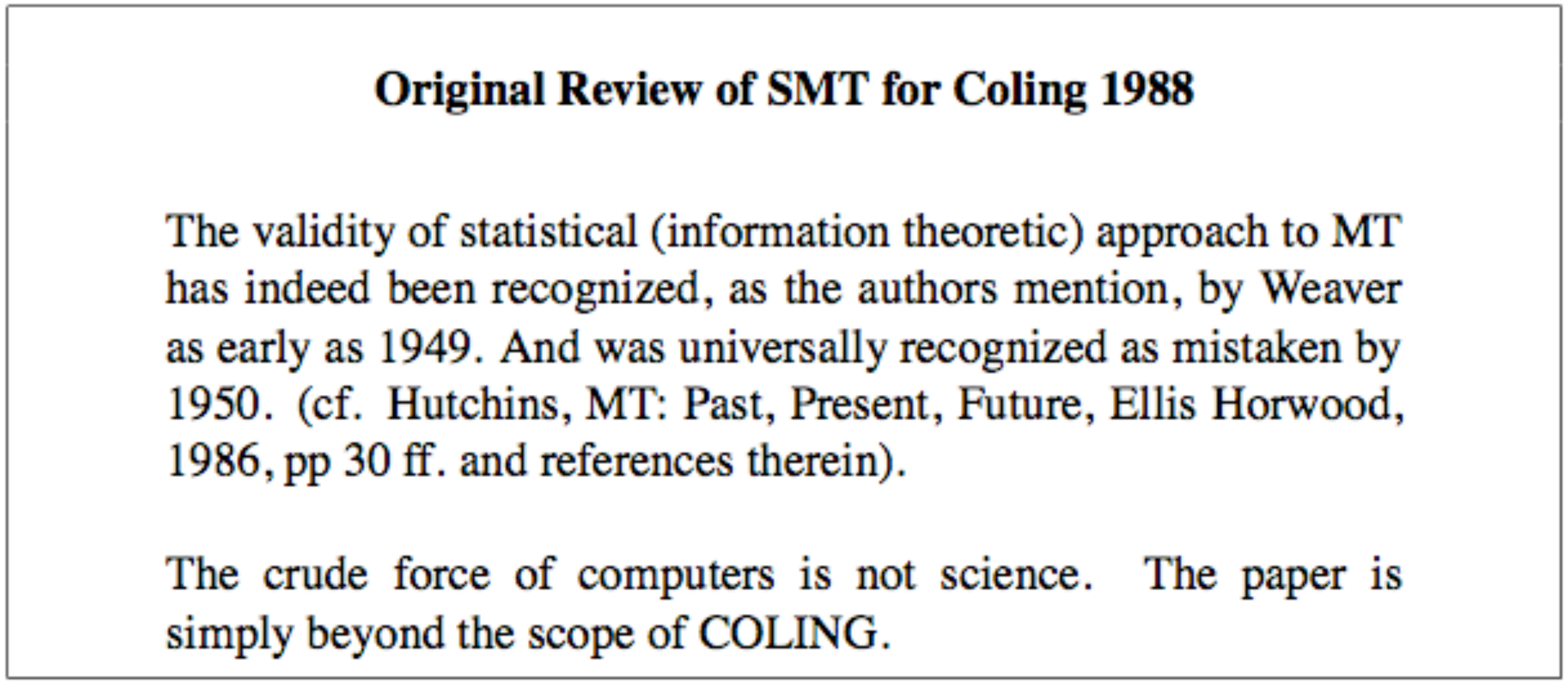
Plot from Church (2007)

UNIVERSITY OF CAMBRIDGE

Some History

- 1950s cognitive science dominated by empiricism (Shannon, Skinner, Harris)
- 1960s and 70s dominated by rationalism (Chomsky, Minsky)
- 1990s sees a return to empiricism (IBM speech group, PDP, neural networks)
- 2015: ML/NNs dominant -- with the recognition of the importance of (structured) knowledge?

Statistical NLP not Science?



Original Review of SMT for Coling 1988

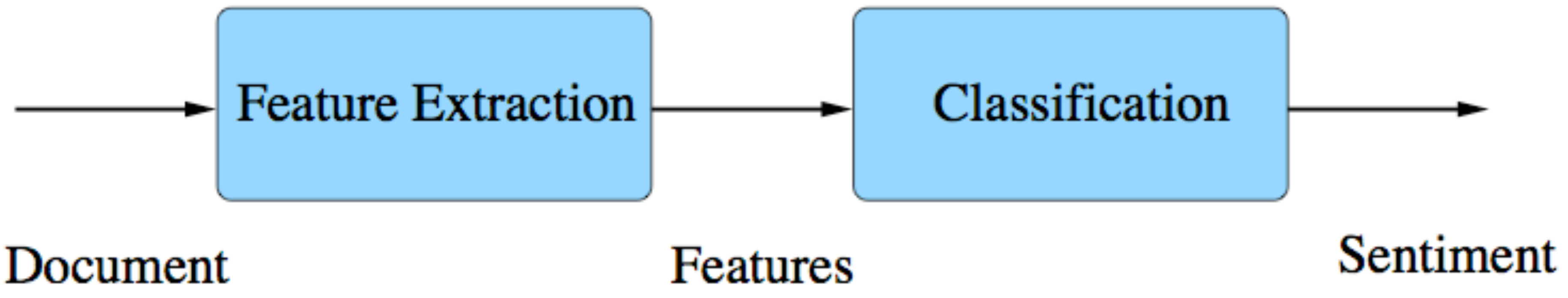
The validity of statistical (information theoretic) approach to MT has indeed been recognized, as the authors mention, by Weaver as early as 1949. And was universally recognized as mistaken by 1950. (cf. Hutchins, MT: Past, Present, Future, Ellis Horwood, 1986, pp 30 ff. and references therein).

The crude force of computers is not science. The paper is simply beyond the scope of COLING.

Text Classification

- Many NLP tasks can be framed as simple classification tasks
- The input is some linguistic unit (word, sentence, document) and the output is a discrete label from some finite set
- Examples include text classification, sentiment analysis, spam detection, ...

Machine Learning Framework



- There are two stages in a pattern recognition framework:
  - feature extraction: a feature vector, $\boldsymbol{x}$, is derived from the "observations";
  - classification: a class ω is identified given the feature vector $\boldsymbol{x}$.
- Example: sentiment analysis
  - $\boldsymbol{w}$ is the document (words)
  - $\boldsymbol{x}$ is a binary vector indicating whether a particular word is in the document
  - ω is the sentiment (e.g. angry)

UNIVERSITY OF CAMBRIDGE

Training and Test Data

- The basic machine learning framework has two sets of data:

  1. Training data: is used to train the classifier - data may be:
     - supervised: the correct classes of the training data are known
     - unsupervised: the correct classes of the training data are not known
     - reinforcement learning: don't learn a model - directly learn an action!
  2. Test data: held-out data for evaluating the classifier

  Supervised training data will be mostly considered in this course

- It is important that the training and test data do not overlap

  - performance on training data better than on held-out data
  - becomes more important as the classifiers become more complex
  - development data sometimes used to tune parameters

- Aim to build a classifier that performs well on held-out data; generalise.

UNIVERSITY OF CAMBRIDGE

Machine Learning-based Decisions

- Consider a system where
  - observation: feature vector of dimension d, $\boldsymbol{x}$
  - class labels: there are K classes, denoted by ω_1, $\omega_2, ...,$ ω_K.
- Classifiers for making decisions can be broadly split as:
  - Generative models: a model of the joint distribution of observations and classes is trained, $P(\boldsymbol{x}, \omega_j)$.
  - Discriminative models: a model of the posterior distribution of the class given the observation is trained, $P(\omega_j|\boldsymbol{x})$.
  - Discriminant functions: a mapping from an observation $\boldsymbol{x}$ to class ω_j is directly trained. No posterior probability, $P(\omega_j|\boldsymbol{x})$, generated just class labels.

UNIVERSITY OF CAMBRIDGE

Naive Bayes Classifier

Suppose there are K classes, $c_1, c_2, c_3, \ldots, c_K$

e.g. $c_1 = \text{politics}, c_2 = \text{sport}, c_3 = \text{cookery}, \ldots$

$$P(c_j|\boldsymbol{x}) = \frac{P(\boldsymbol{x}|c_j)P(c_j)}{P(\boldsymbol{x})}$$

where $\boldsymbol{x}$ is the feature vector for the input document

$$P(c_j|\boldsymbol{x}) \propto P(\boldsymbol{x}|c_j)P(c_j)$$

UNIVERSITY OF CAMBRIDGE

Bag of Words Model

$$P(\boldsymbol{x}|c_j) = P(f_1, f_2, f_3, \ldots, f_N|c_j)$$

where f_i is a binary-valued indicator function for ith word

Assume words are conditionally independent given class:

$$P(f_1, f_2, f_3, \ldots, f_N|c_j) = \prod_{i=1}^{N} P(f_i|c_j)$$

Probability Estimation

$$\hat{P}(f_i|c_j) = \frac{\text{freq}(f_i, c_j)}{\text{freq}(c_j)}$$

$$\hat{P}(c_j) = \frac{\text{freq}(c_j)}{number\ of\ docs}$$

relative frequency estimates for this model are *maximum likelihood estimates*

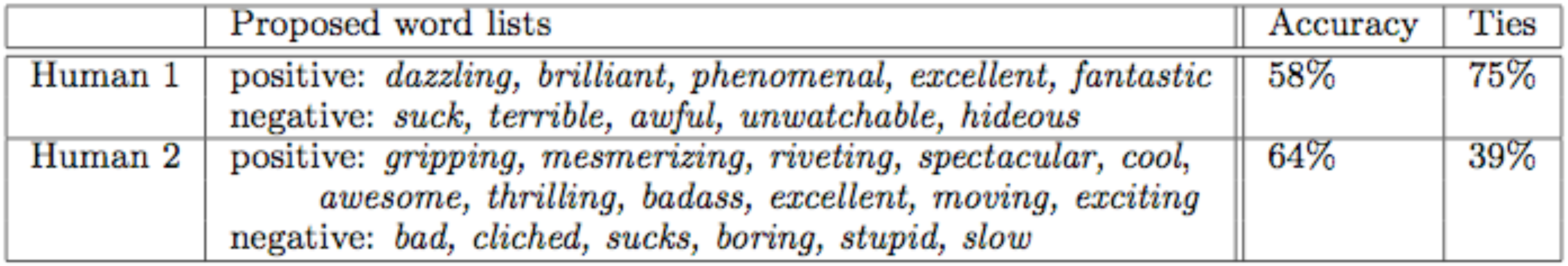
UNIVERSITY OF CAMBRIDGE

Sentiment Classification

- Pang and Lee started the NLP literature in 2002
- Movie reviews online (IMDb) provide a readily available set of annotated training data
- Classes *positive*, *negative (*752 -ve, 1301 +ve)

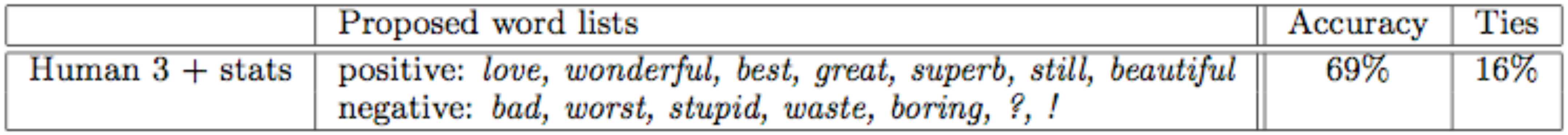
UNIVERSITY OF CAMBRIDGE

Bag of Words Model



	Proposed word lists	Accuracy	Ties
Human 1	positive: *dazzling, brilliant, phenomenal, excellent, fantastic* negative: *suck, terrible, awful, unwatchable, hideous*	58%	75%
Human 2	positive: *gripping, mesmerizing, riveting, spectacular, cool, awesome, thrilling, badass, excellent, moving, exciting* negative: *bad, cliched, sucks, boring, stupid, slow*	64%	39%

Figure 1: Baseline results for human word lists. Data: 700 positive and 700 negative reviews.



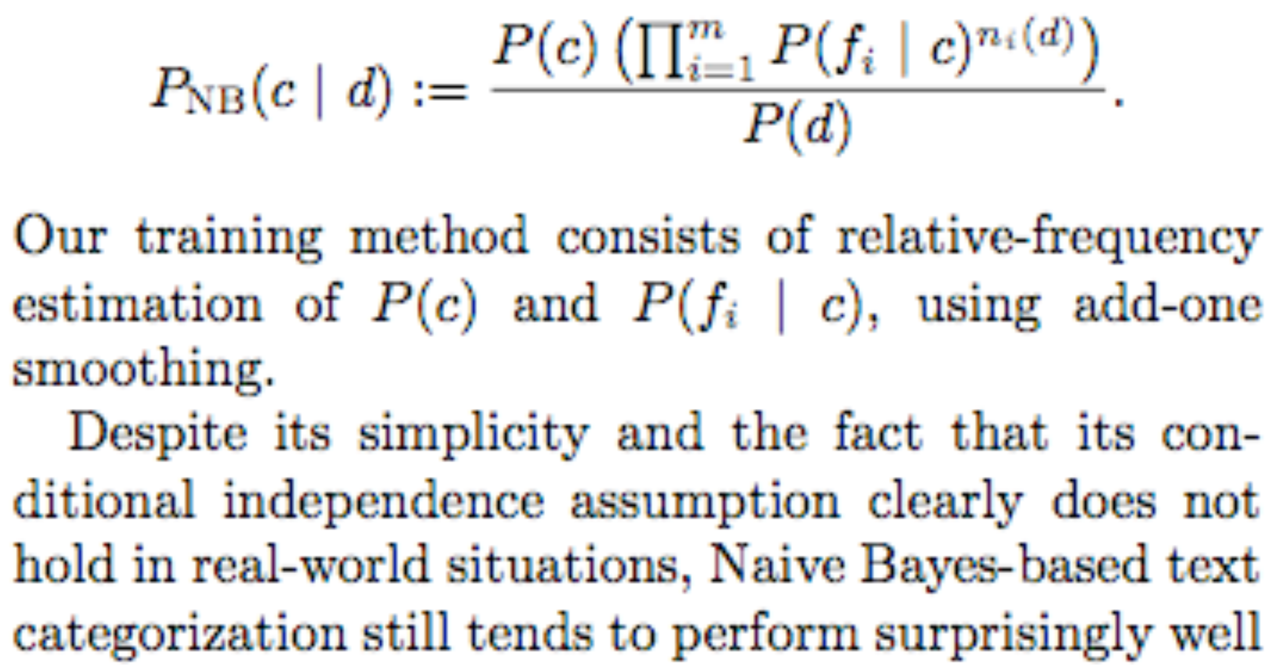
	Proposed word lists	Accuracy	Ties
Human 3 + stats	positive: *love, wonderful, best, great, superb, still, beautiful* negative: *bad, worst, stupid, waste, boring, ?, !*	69%	16%

Figure 2: Results for baseline using introspection and simple statistics of the data (including *test* data).

taken from Pang et al. (2002)

UNIVERSITY OF CAMBRIDGE

Bag of Words Model



$$P_{\mathrm{NB}}(c \mid d) := \frac{P(c)\left(\prod_{i=1}^{m} P(f_i \mid c)^{n_i(d)}\right)}{P(d)}.$$

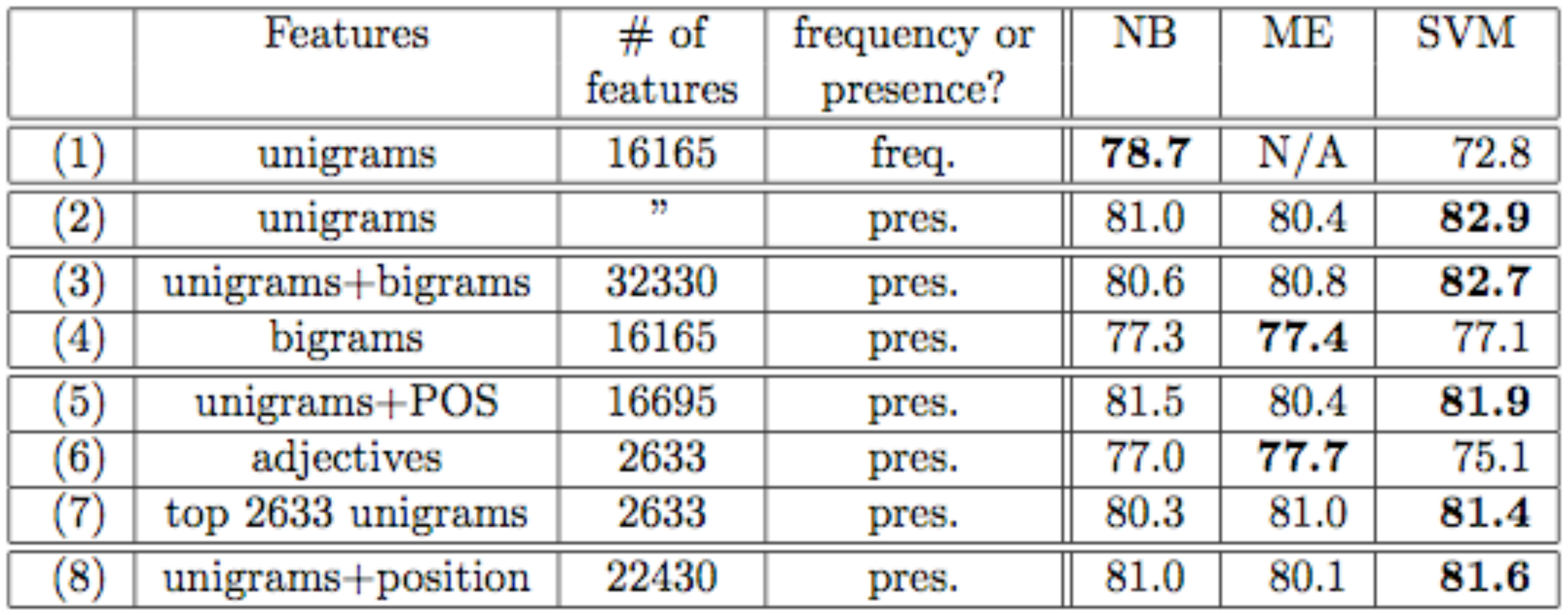
Our training method consists of relative-frequency estimation of $P(c)$ and $P(f_i \mid c)$, using add-one smoothing.

Despite its simplicity and the fact that its conditional independence assumption clearly does not hold in real-world situations, Naive Bayes-based text categorization still tends to perform surprisingly well

taken from Pang et al. (2002)

UNIVERSITY OF CAMBRIDGE

Results



	Features	# of features	frequency or presence?	NB	ME	SVM
(1)	unigrams	16165	freq.	**78.7**	N/A	72.8
(2)	unigrams	"	pres.	81.0	80.4	**82.9**
(3)	unigrams+bigrams	32330	pres.	80.6	80.8	**82.7**
(4)	bigrams	16165	pres.	77.3	**77.4**	77.1
(5)	unigrams+POS	16695	pres.	81.5	80.4	**81.9**
(6)	adjectives	2633	pres.	77.0	**77.7**	75.1
(7)	top 2633 unigrams	2633	pres.	80.3	81.0	**81.4**
(8)	unigrams+position	22430	pres.	81.0	80.1	**81.6**

Figure 3: Average three-fold cross-validation accuracies, in percent. Boldface: best performance for a given setting (row). Recall that our baseline results ranged from 50% to 69%.

taken from Pang et al. (2002)

UNIVERSITY OF CAMBRIDGE

More Recent Work on Sentiment

- Now a very large literature! (and commercial interest)
- See Richard Socher’s work using a form of recursive neural network
- Also see TheySay, a recent spin out from Oxford (http://www.theysay.io/)