A Sentimental Education: Sentiment Analysis Using Subjectivity Summarization Based on Minimum Cuts

Bo Pang and Lillian Lee (2004)

Document-level Polarity Classification

- Determining whether an article is a good or bad movie review
- Resistant to data-driven methods (counting positive, negative words)
- A lot of the text is objective (plot summary, etc.)

Sentence-level Subjectivity Extraction

- Polarity classification would be easier if you could eliminate the plot summaries
- Classify sentences as objective or subjective, throw out the objective ones and then classify what's left
- How?

Sentence-level Subjectivity Extraction

- You could come up with some interesting features and train a classifier with those.
- But this is a paper about graph-based models!

Pairwise interaction information

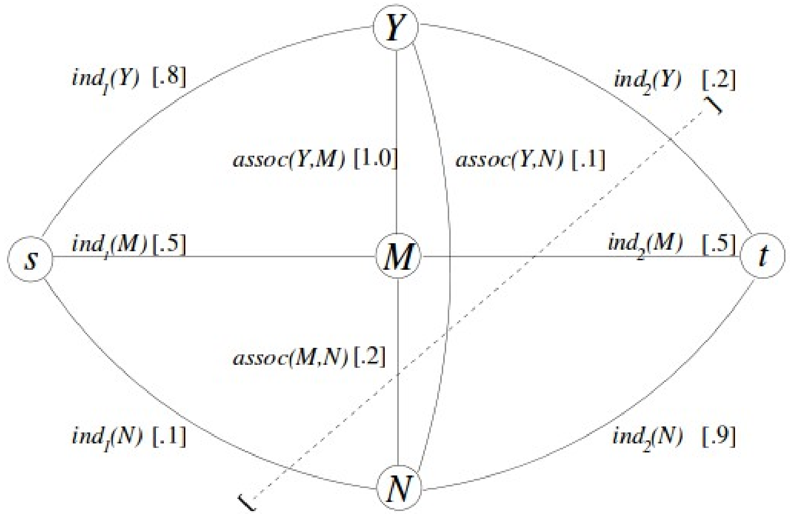
- You want individual feature vectors for each sentence $ind_j(x_i)$
- you also want to measure how important it is that two sentences belong to the same class, never mind which one. Call those $assoc(x_i, x_k)$
- Minimize this:

$$\sum_{x \in C_1} ind_2(x) + \sum_{x \in C_2} ind_1(x) + \sum_{\substack{x_i \in C_1, \\ x_k \in C_2}} assoc(x_i, x_k).$$

The graph part

- Cut of a graph: a partition of the vertices of a graph into two disjoint subsets that are joined by at least one edge (wikipedia)
- Minimum cut: the cut such that the edges that separate the subsets have minimum weight
- If you set it up right, you can use it to minimize the equation

Setting up the graph



C_1	Individual penalties	Association penalties	Cost
{Y,M}	.2 + .5 + .1	.1 + .2	1.1
(none)	.8 + .5 + .1	0	1.4
{Y,M,N}	.2 + .5 + .9	0	1.6
{Y}	.2 + .5 + .1	1.0 + .1	1.9
{N}	.8 + .5 + .9	.1 + .2	2.5
{M}	.8 + .5 + .1	1.0 + .2	2.6
{Y,N}	.2 + .5 + .9	1.0 + .2	2.8
{M,N}	.8 + .5 + .9	1.0 + .1	3.3

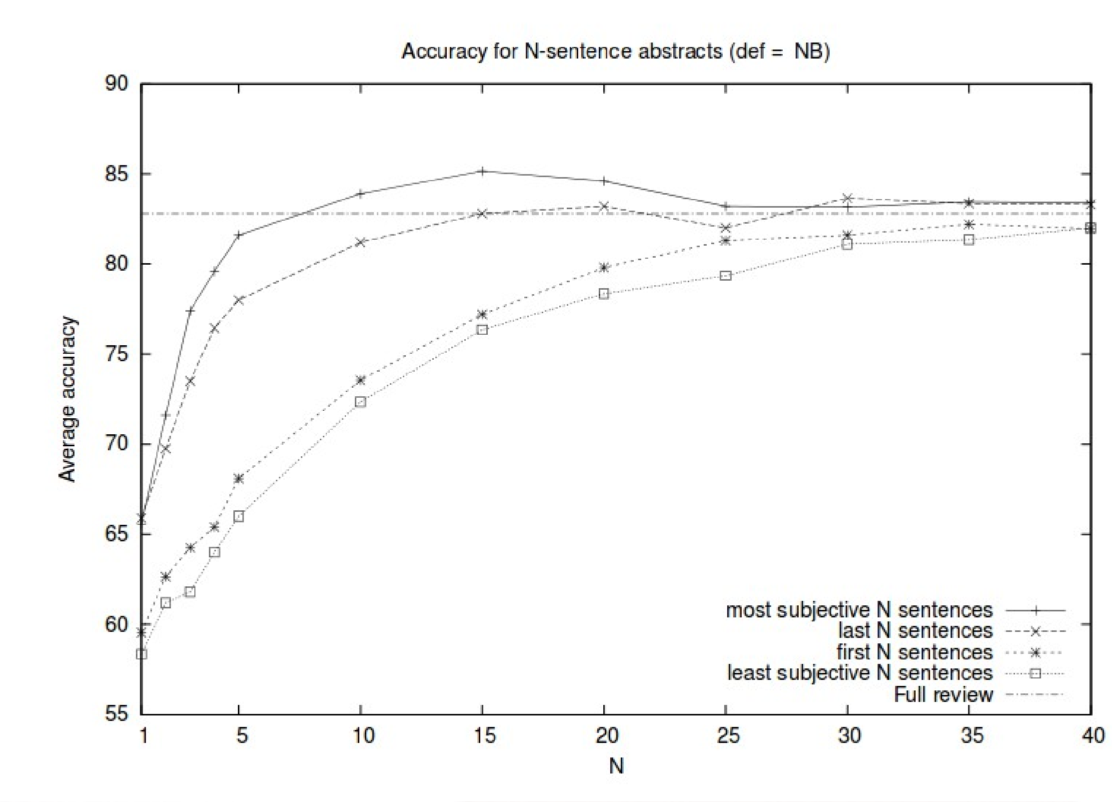
The data

- Polarity dataset: 2000 reviews, half positive and half negative, max 20 per author
- Subjectivity dataset: 5000 review snippets from rottentomatoes, 5000 plot summary snippets from imdb, collected automatically

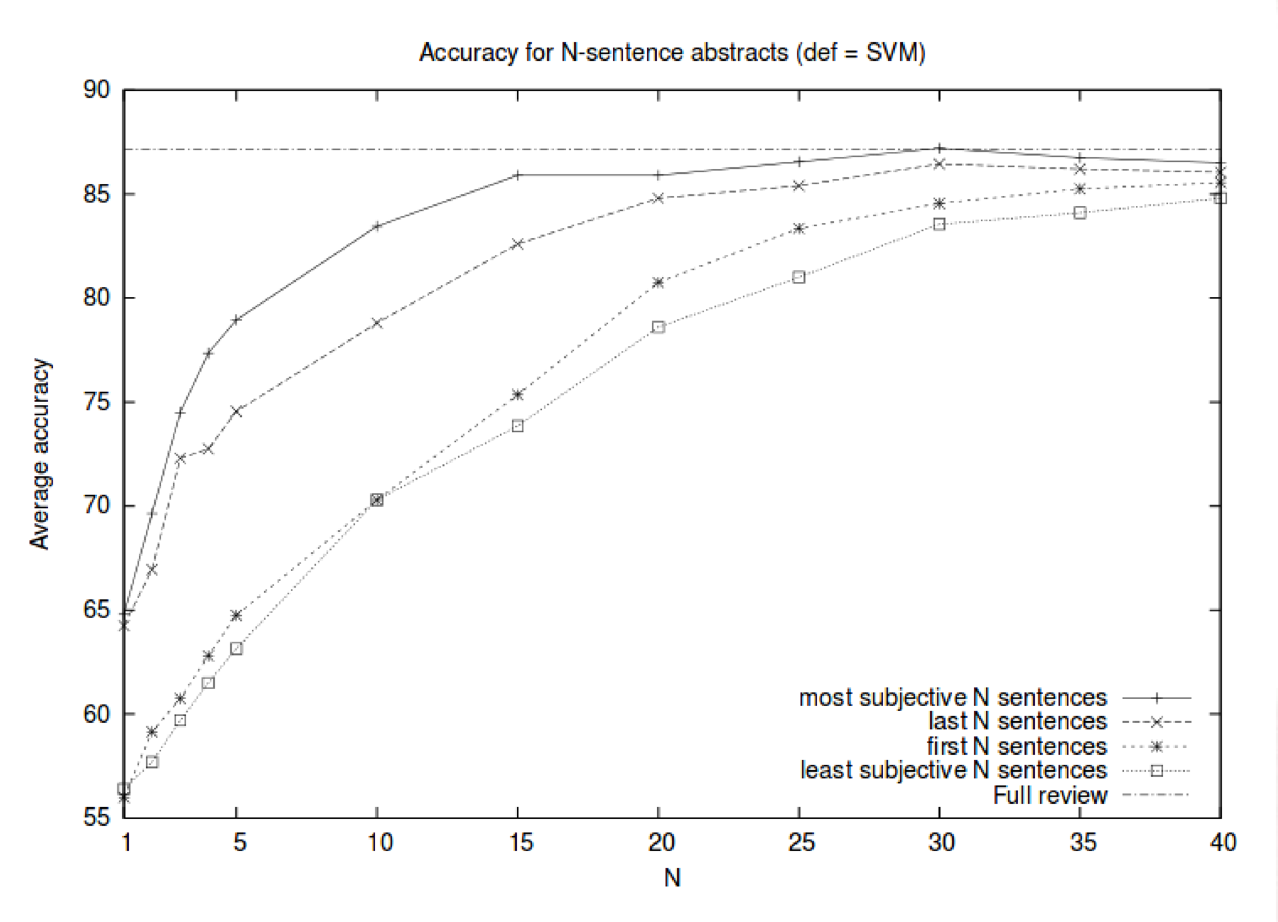
Experiments – no minimum cut

- Train a polarity classifier on the polarity dataset. Use unigram presence features, and do 10-fold cross-evaluation.
- Classify based on the full review, the first N, and the last N sentences with various values of N.
- Do subjectivity detection without also considering proximity (no graph models yet). Train classifiers on the subjectivity dataset. Extract the N most subjective sentences.
- Also try with the N least subjective

Results – no minimum cut

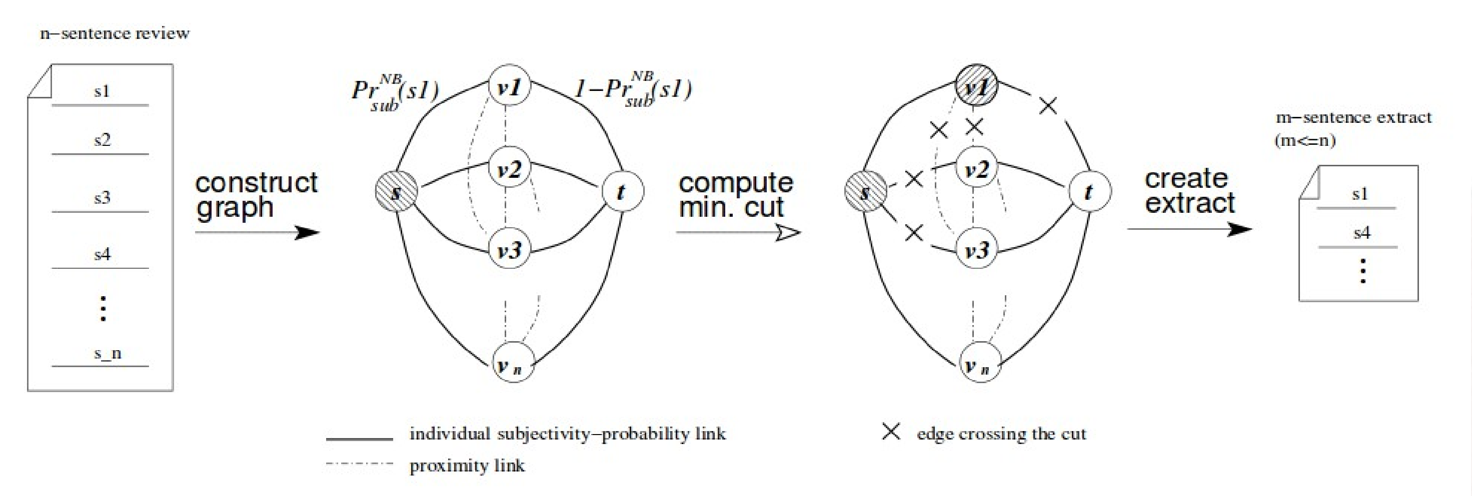


Results – no minimum cut

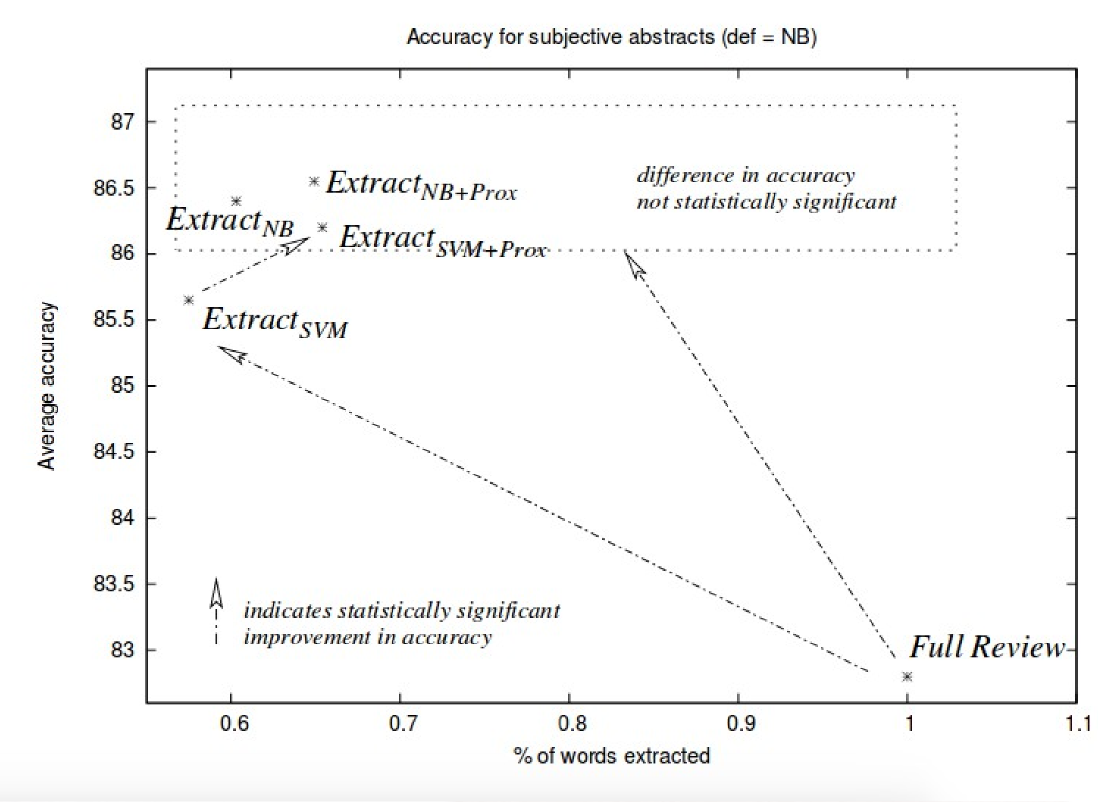


Experiments – minimum cut

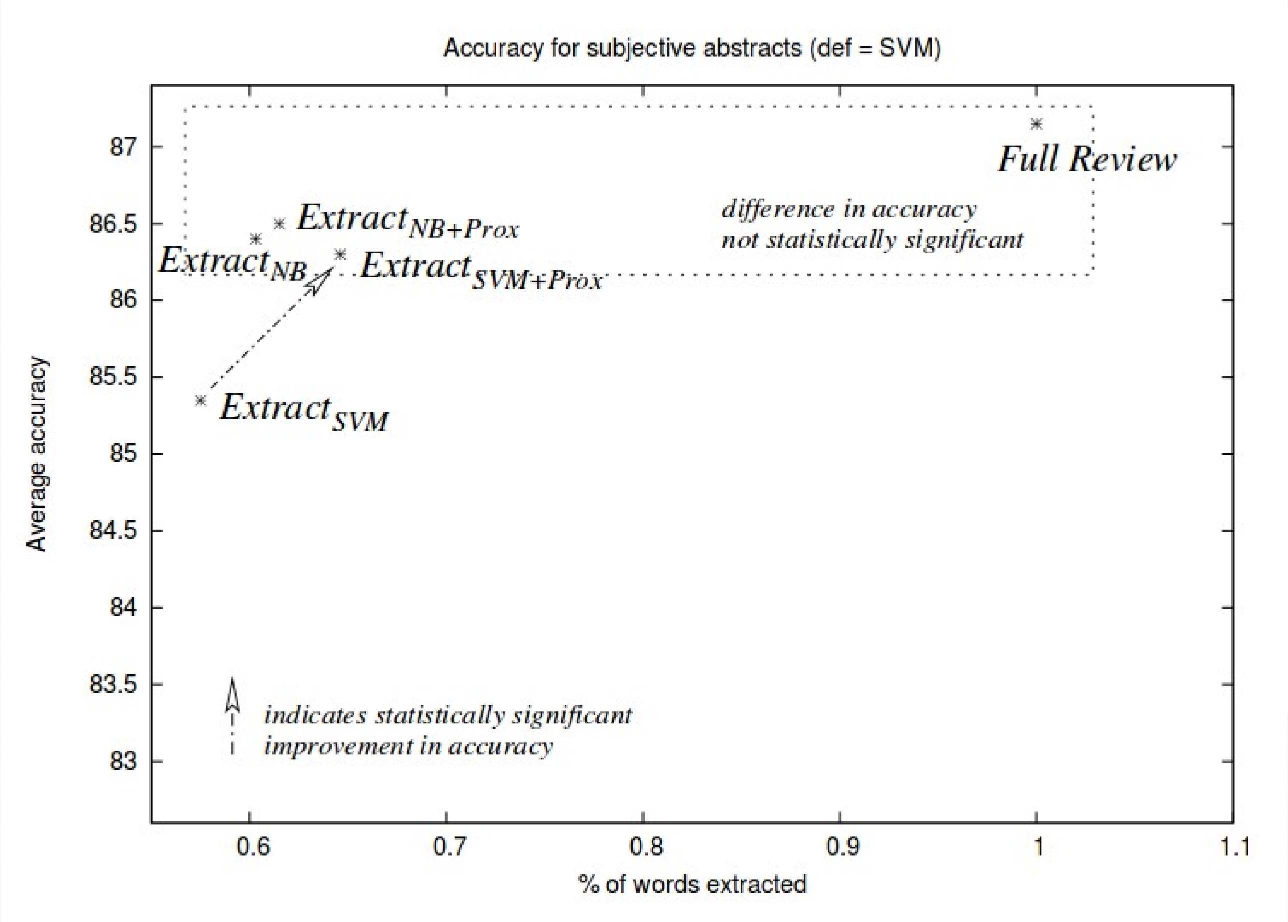
- In addition to the individual subjectivity scores for sentences, give them proximity scores to the other sentences in the same document.
- Find the minimum cut, extract the N most subjective again.



Results – minimum cut



Results – minimum cut



Learning General Connotations of Words using Graph-based Algorithms

- Song Feng, Ritwik Bose, Yejin Choi

Problem

- Sentiment Lexicons
- Connotation Lexicons
  - World knowledge?
  - Connotative predicates

Connotative Predicates

- Selectional preference of connotative predicates
- Example: prevent, congratulate
- Semantic prosody

Connotation

- Some words have polar connotation even though they are objective
- Predicates are not necessarily words with strong sentiment and inverse
- Ex's: save, illuminate, cause, abandon

Creating a Graph

- Predicates on left, words with connotative polarity on right, thickness of edges is strength of association
- Only look at THEME role of predicate
- Given seed predicates, learn connotation lexicon and new predicates via graph centrality

Graphs

- Two types: undirected (symmetric) and directed (asymmetric)
- Different edge weighting: PMI and conditional probability
- Start with seed of specifically connotative predicates

HITS

- Good hubs point to many good authorities, good authorities pointed to by many good hubs
- Authority and hub scores calculated recursively
- $a(Ai)=\sum_{Pi,\, Aj \in E} w(i,j)h(Aj)+ \sum_{Pj,\, Ai \in E} h(Pj)w(j,i)$
- $h(Ai)= \sum_{Pi,\, Aj \in E} w(i,j)a(Aj)+ \sum_{Pj,\, Ai \in E} a(Pj)w(j,i)$

PageRank

- Based on edges leading into and out of nodes, which are either predicates or arguments
- $S(i) = \alpha \sum_{j \in In(i)} S(j) \times w(i, j)/|Out(i)| + (1 - \alpha)$

Tests

- Both symmetric and asymmetric graphs
- Both truncated and focused (teleportation)
- Data from Google Web 1T
- Co-occurrence pattern: [p] [*]ˆn-2 [a]

Comparison to Sentiment Lexicons

- Compare overlap with two sentiment lexicons: General Inquirer and Opinion Finder
- Best results
  - General Inquirer 73.6 vs 77.7
  - Opinion Finder 83.0 vs 86.3

Extrinsic Evaluation via Sentiment Analysis

- Evaluated on SemEval2007 and Sentiment Twitter
- BOW + Opinion Finder + connotation lexicon
- 78.0 vs 71.4 on Sentiment Twitter

Intrinsic Evaluation via Human Judgment

- Human judges give connotative polarity judgments for words (1-5)
- 97% on control, 94% on words without graph, 87.3 vs 79.8 for graph words

Critique

- Solution in search of problem?
- No discussion of low human evaluation score
- Comparison with sentiment lexicons may not be informative – idea is to find words NOT in lexicons
- Naive predicate/argument extraction - very confident that noise will be filtered out

Positives

- Connotation lexicon seems intuitively important
- Graph algorithms are great work-arounds to world knowledge-heavy task
- Uses theoretically motivated linguistic knowledge and find results