A Brief History of the Short History of Text Mining (in Finance)

Sanjiv Ranjan Das and Karthik Mokashi

Reference monograph

Text expands the universe of data by many-fold. See my monograph on text mining in finance at:
http://algo.scu.edu/~sanjivdas/Das_TextAnalyticsInFinance.pdf

This covers some of the content of this presentation.

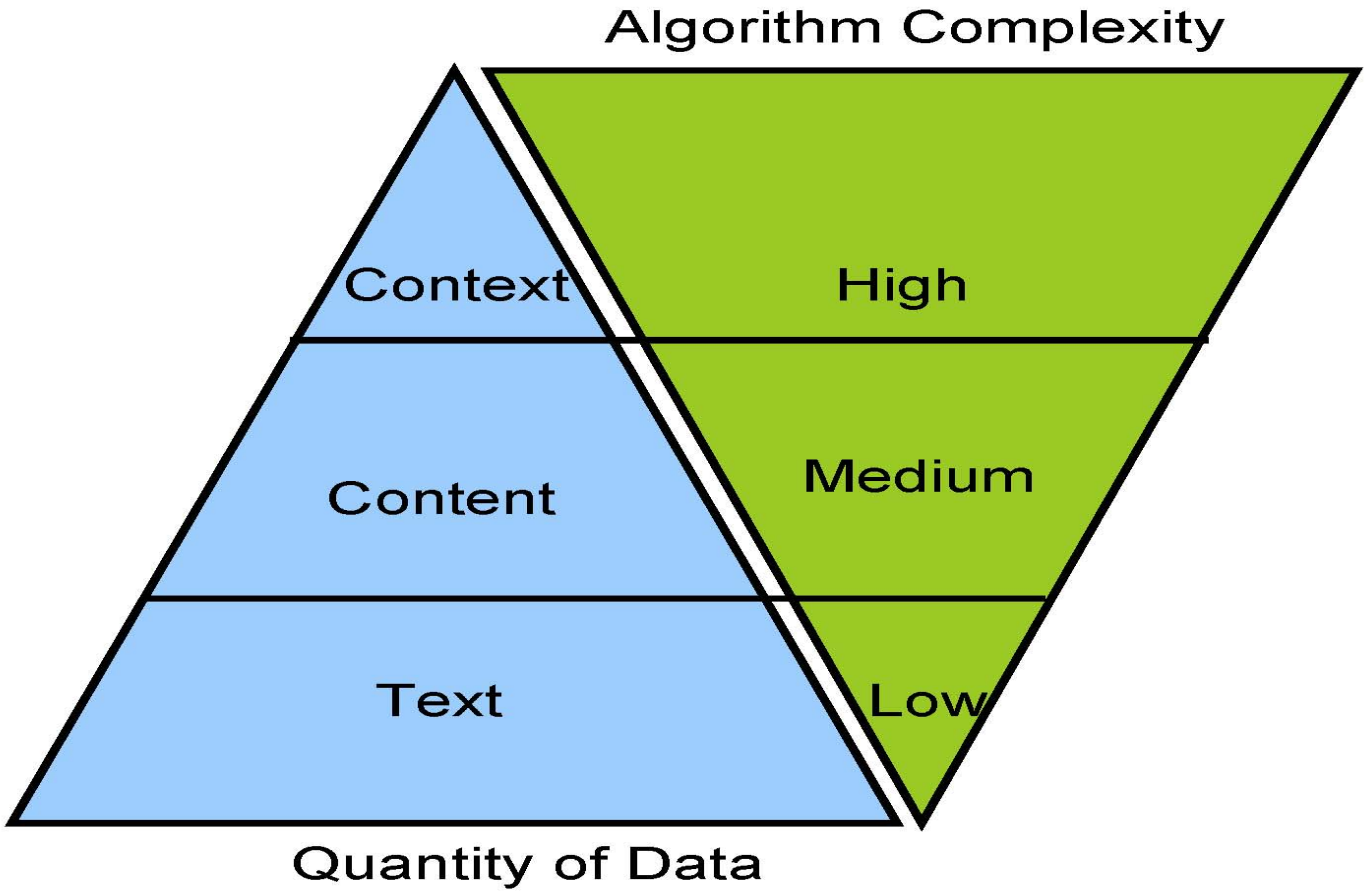
Text as Data

1. Big Text: there is more textual data than numerical data.
2. Text is versatile. Nuances and behavioral expressions that are not conveyed with numbers.
3. Text contains emotive content. Sentiment analysis. Admati-Pfleiderer 2001; DeMarzo et al 2003; Antweiler-Frank 2004, 2005; Das-Chen 2007; Tetlock 2007; Tetlock et al 2008; Mitra et al 2008; Leinweber-Sisk 2010.
4. Text contains opinions and connections. Das et al 2005; Das and Sisk 2005; Godes et al 2005; Li 2006; Hochberg et al 2007.
5. Numbers aggregate; text categorizes (topics).
6. Text can be generated.

Definition: Text-Mining

1. Text mining is the large-scale, automated processing of plain text language in digital form to extract data that is converted into useful quantitative or qualitative information.
2. Text mining is automated on big data that is not amenable to human processing within reasonable time frames. It entails extracting data that is converted into information of many types.
3. Simple: Text mining may be simple as in key word searches and counts.
4. Complicated: It may require language parsing and complex rules for information extraction.
5. Structured text, such as the information in forms and some kinds of web pages.
6. Unstructured text is a much harder endeavor.

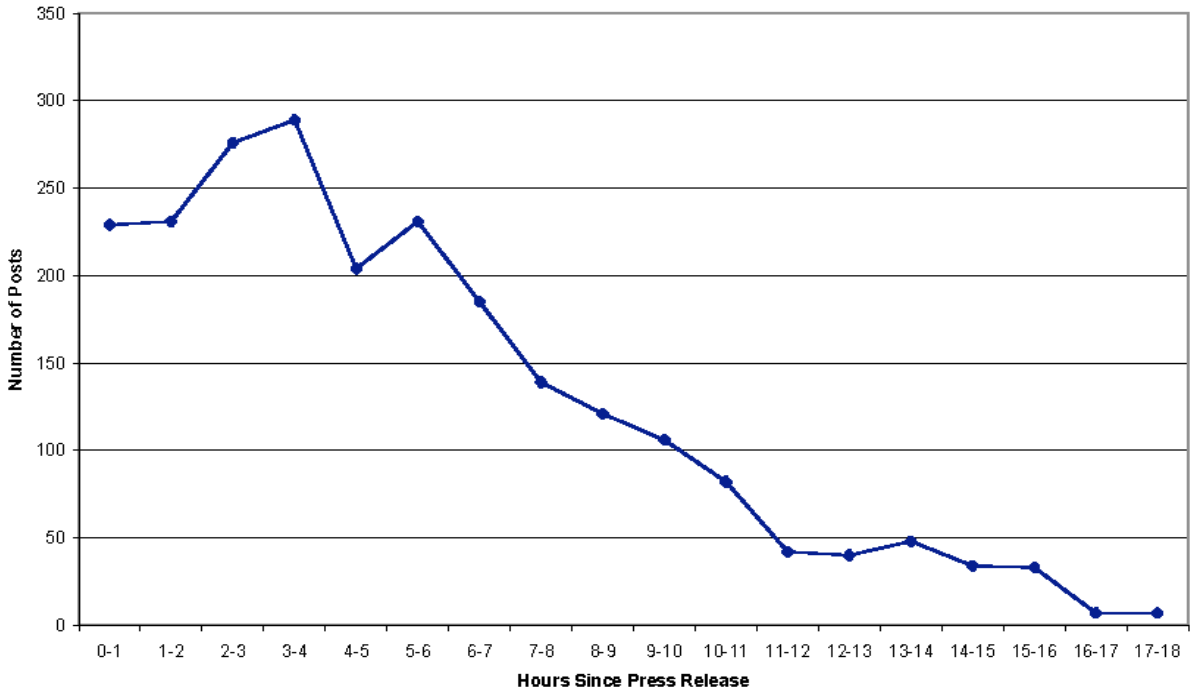
Data and Algorithms



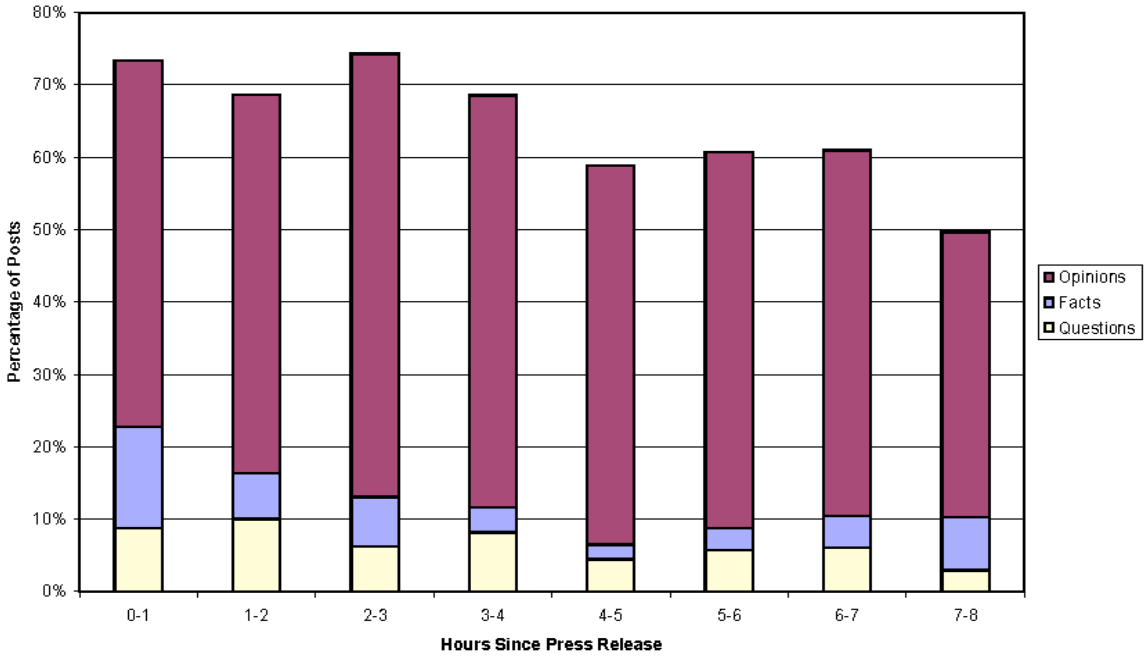
Text Handling

- Text Extraction
- String Parsing
- String Detection
- Text cleanup
- XML Package
- Regular Expressions
- Using APIs (Twitter, Facebook, Yelp)

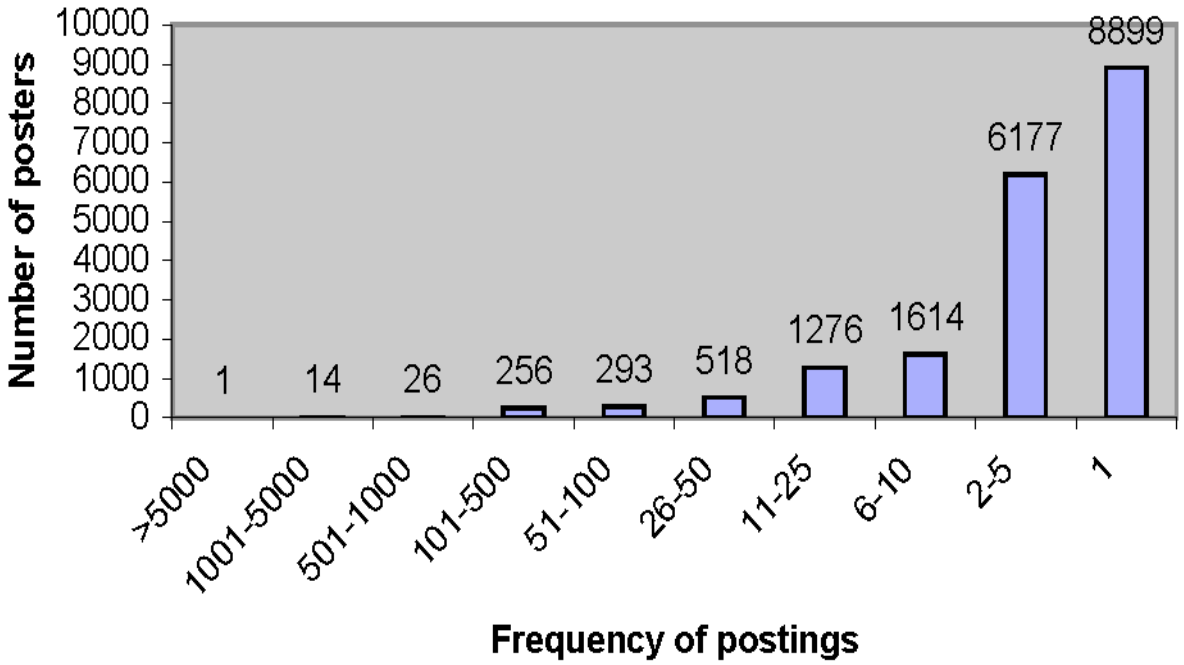
The Response to News (Das, Martinez-Jerez, and Tufano (FM 2005))



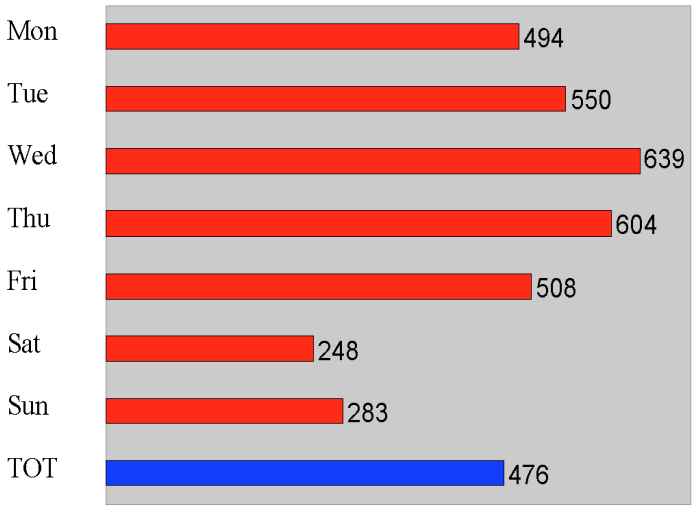
Breakdown of News Flow

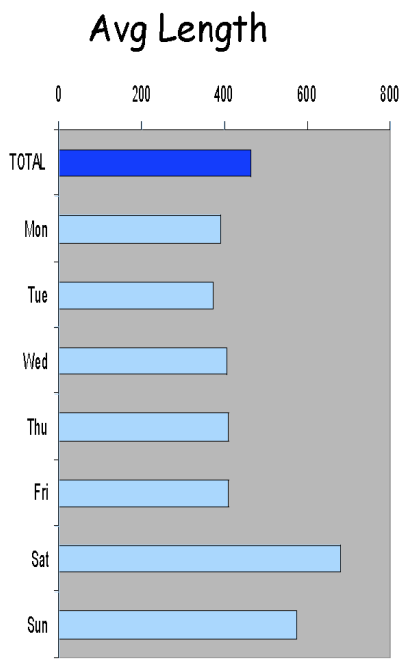


Frequency of Postings

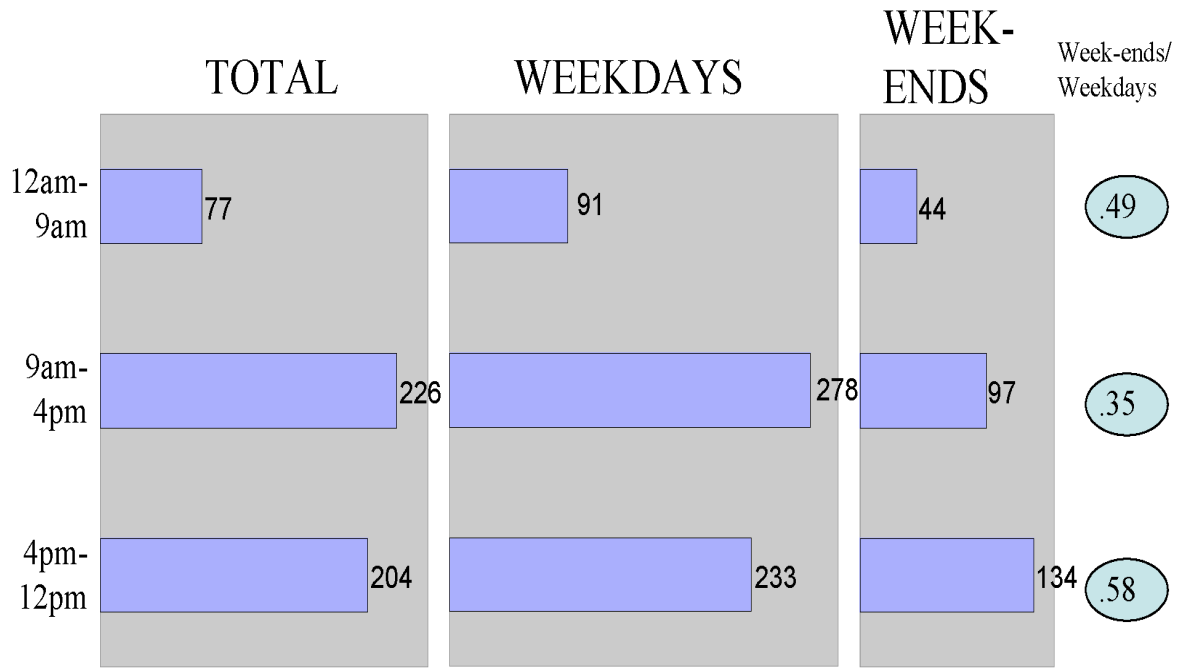


Weekly Posting

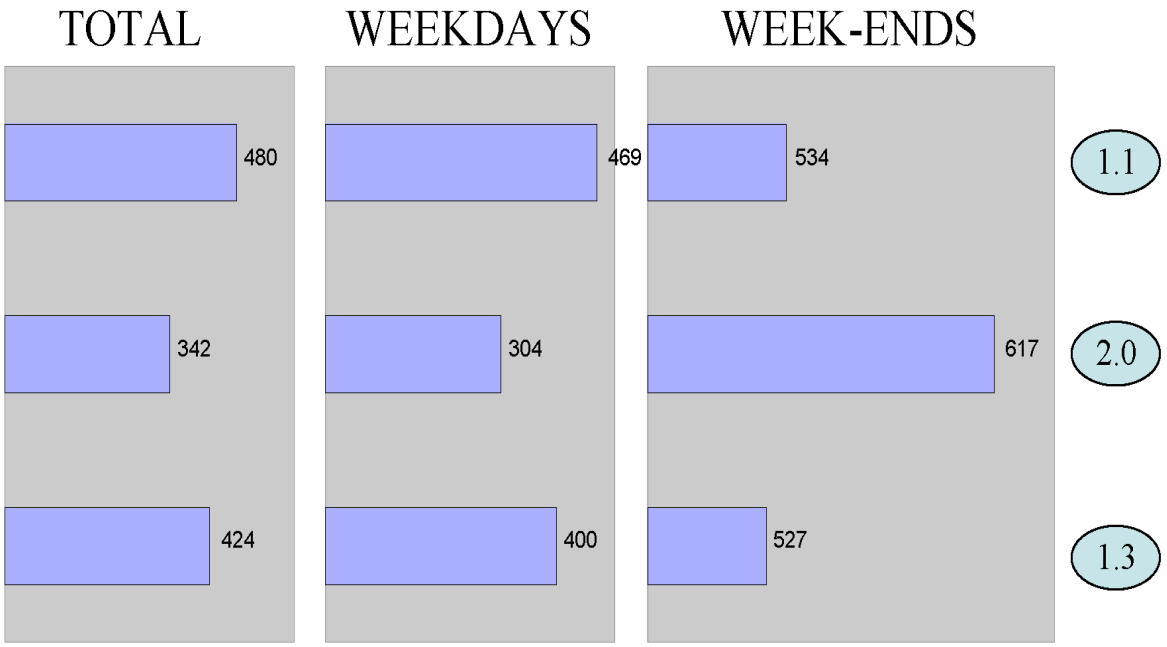




Intraday Posting



Number of Characters per Posting



Text Mining with the "tm" Package

1. Functions such as {readDOC()}, {readPDF()}, etc., for reading DOC and PDF files, the package makes accessing various file formats easy.

2. Text mining involves applying functions to many text documents, a **corpus**. Ability to operate on the entire set of documents at one go.

3. Remove Stopwords, Punctuation, Numbers. Create a Bag of Words (BOW)

4. Term Document Matrix (TDM): An entire library of text inside a single matrix. Analysis and searching documents. In search engines, topic analysis, and classification (spam filtering).

5. Term Frequency - Inverse Document Frequency (TF-IDF)

Wordclouds

Wordlcouds are interesting ways in which to represent text. They give an instant visual summary. The **wordcloud** package in R may be used to create your own wordclouds.

Document Similarity

In this segment we will learn some popular functions on text that are used in practice. One of the first things we like to do is to find similar text or like sentences (think of web search as one application). Since documents are vectors in the TDM, we may want to find the closest vectors or compute the distance between vectors.

$$cos(\theta) = \frac{A \cdot B}{||A|| \times ||B||}$$

where $||A|| = \sqrt{A \cdot A}$, is the dot product of A with itself, also known as the norm of A. This gives the cosine of the angle between the two vectors and is zero for orthogonal vectors and 1 for identical vectors.

Dictionaries

1. Standard Dictionaries: www.dictionary.com, and www.merriam-webster.com.
2. The Harvard General Inquirer: http://www.wjh.harvard.edu/~inquirer/
3. Computer dictionary: http://www.hyperdictionary.com/computer that contains about 14,000 computer related words, such as "byte" or "hyperlink".
4. Math dictionary, such as http://www.amathsdictionaryforkids.com/dictionary.html.
5. Medical dictionary, see http://www.hyperdictionary.com/medical.

Dictionaries - II

1. Internet lingo dictionaries. http://www.netlingo.com/dictionary/all.php for words such as "2BZ4UQT" which stands for "too busy for you cutey" (LOL).

2. Associative dictionaries are also useful when trying to find context, as the word may be related to a concept, identified using a dictionary such as http://www.visuwords.com/. This dictionary doubles up as a thesaurus.

3. Value dictionaries deal with values and may be useful when only affect (positive or negative) is insufficient for scoring text. The Lasswell Value Dictionary http://www.wjh.harvard.edu/~inquirer/lasswell.htm may be used to score the loading of text on the eight basic value categories: Wealth, Power, Respect, Rectitude, Skill, Enlightenment, Affection, and Well being.

Lexicons

1. A **lexicon** is defined by Webster's as "a book containing an alphabetical arrangement of the words in a language and their definitions; the vocabulary of a language, an individual speaker or group of speakers, or a subject; the total stock of morphemes in a language." This suggests it is not that different from a dictionary.

2. A "morpheme" is defined as "a word or a part of a word that has a meaning and that contains no smaller part that has a meaning."

3. In the text analytics realm, we will take a lexicon to be a smaller, special purpose dictionary, containing words that are relevant to the domain of interest.

4. The computational effort required by text analytics algorithms is drastically reduced.

Constructing a lexicon

1. By hand. This is an effective technique and the simplest. It calls for a human reader who scans a representative sample of text documents and culls important words that lend interpretive meaning.

2. Examine the term document matrix for most frequent words, and pick the ones that have high connotation for the classification task at hand.

3. Use pre-classified documents in a text corpus. We analyze the separate groups of documents to find words whose difference in frequency between groups is highest. Such words are likely to be better in discriminating between groups.

Lexicons as Word Lists

1. Das and Chen (2007) constructed a lexicon of about 375 words that are useful in parsing sentiment from stock message boards. This lexicon also introduced the notion of "negation tagging" into the literature.

2. Loughran and McDonald (2011):

- Taking a sample of 50,115 firm-year 10-Ks from 1994 to 2008, they found that almost three-fourths of the words identified as negative by the Harvard Inquirer dictionary are not typically negative words in a financial context.
- Therefore, they specifically created separate lists of words by the following attributes of words: negative, positive, uncertainty, litigious, strong modal, and weak modal. Modal words are based on Jordan's categories of strong and weak modal words. These word lists may be downloaded from http://www3.nd.edu/~mcdonald/Word_Lists.html.

Scoring Text

Mood Scoring using Harvard Inquirer

```
Entryword Source Pos Neg Pstv Affil Ngtv Hostile Strng Power Weak Subm Actv Psv
Pleasure Pain Arousal EMOT Feel Virtue Vice Ovrst Undrst Acad Doctr Econ* Exch E
CON Exprs Legal Milit Polit* POLIT Relig Role COLL Work Ritual Intrel Race Kin*
MALE Female Nonadlt HU ANI PLACE Social Region Route Aquatic Land Sky Object Too
l Food Vehicle Bldgpt Natobj Bodypt Comnobj Comform COM Say Need Goal Try Means
Ach Persist Complt Fail Natpro Begin Vary Change Incr Decr Finish Stay Rise Move
 Exert Fetch Travel Fall Think Know Causal Ought Percv Comp Eval EVAL Solve Abs*
 ABS Qual Quan NUMB ORD CARD FREQ DIST Time* TIME Space POS DIM Dimn Rel COLOR S
elf Our You Name Yes No Negate Intrj IAV DAV SV IPadj IndAdj POWGAIN POWLOSS POW
ENDS POWAREN POWCON POWCOOP POWAPT POWPT POWDOCT POWAUTH POWOTH POWTOT RCTETH RC
TREL RCTGAIN RCTLOSS RCTENDS RCTTOT RSPGAIN RSPLOSS RSPOTH RSPTOT AFFGAIN AFFLOS
S AFFPT AFFOTH AFFTOT WLTPT WLTTRAN WLTOTH WLTTOT WLBGAIN WLBLOSS WLBPHYS WLBPSY
C WLBPT WLBTOT ENLGAIN ENLLOSS ENLENDS ENLPT ENLOTH ENLTOT SKLAS SKLPT SKLOTH SK
LTOT TRNGAIN TRNLOSS TRANS MEANS ENDS ARENAS PARTIC NATIONS AUD ANOMIE NEGAFF PO
SAFF SURE IF NOT TIMESP FOOD FORM Othertags Definition
A H4Lvd DET ART  | article: Indefinite singular article--some or any one
ABANDON H4Lvd Neg Ngtv Weak Fail IAV AFFLOSS AFFTOT SUPV  |
ABANDONMENT H4 Neg Weak Fail Noun  |
ABATE H4Lvd Neg Psv Decr IAV TRANS SUPV  |
ABATEMENT Lvd Noun
ABDICATE H4 Neg Weak Subm Psv Finish IAV SUPV  |
ABHOR H4 Neg Hostile Psv Arousal SV SUPV  |
ABIDE H4 Pos Affil Actv Doctr IAV SUPV  |
ABIDE#1 Lvd Modif
ABIDE#2 Lvd SUPV
ABILITY Lvd MEANS Noun ABS ABS*
ABJECT H4 Neg Weak Subm Psv Vice IPadj Modif  |
ABLE H4Lvd Pos Pstv Strng Virtue EVAL MEANS Modif  | adjective: Having necessary
 power, skill, resources, etc.
ABNORMAL H4Lvd Neg Ngtv Vice NEGAFF Modif  |
ABOARD H4Lvd Space PREP LY  |
ABOLISH H4Lvd Neg Ngtv Hostile Strng Power Actv Intrel IAV POWOTH POWTOT SUPV  |
ABOLITION Lvd TRANS Noun
```

Language Detection

We may be scraping web sites from many countries and need to detect the language and then translate it into English for mood scoring. The useful package **textcat** enables us to categorize the language.

```
library(textcat)
text = c("Je suis un programmeur novice.",
         "I am a programmer who is a novice.",
         "Sono un programmatore alle prime armi.",
         "Ich bin ein Anfänger Programmierer",
         "Soy un programador con errores.")

lang = textcat(text)
print(lang)
```

```
## [1] "french"  "english" "italian" "german"  "spanish"
```

Language Translation

```
library(translate)
set.key("AIzaSyDIB8qQTmhLlbPNN38Gs4dXnlN4a7lRrHQ")
print(translate(text[1],"fr","en"))
```

```
## [[1]]
## [1] "I am a novice programmer."
```

```
print(translate(text[3],"it","en"))
```

```
## [[1]]
## [1] "I&#39;m a novice programmer."
```

```
print(translate(text[4],"de","en"))
```

```
## [[1]]
## [1] "I am a beginner programmer"
```

```
print(translate(text[5],"es","en"))
```

Text Classification

1. Machine classification is, from a layman's point of view, nothing but learning by example. In new-fangled modern parlance, it is a technique in the field of "machine learning".

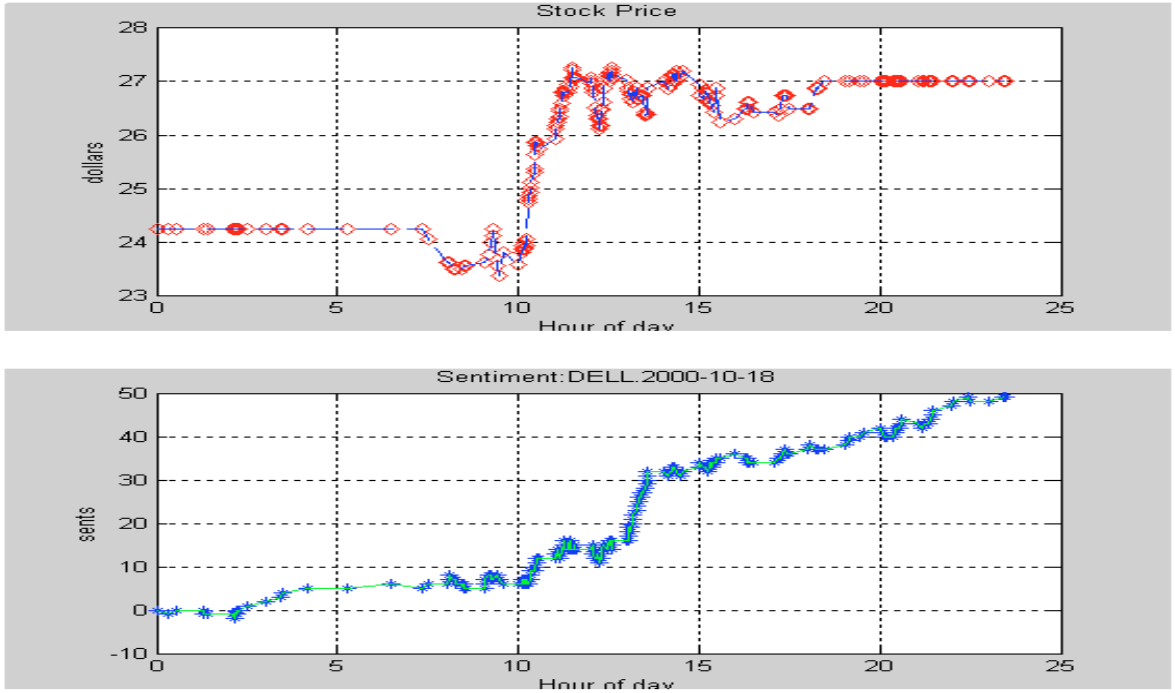
2. Learning by machines falls into two categories, supervised and unsupervised. When a number of explanatory X variables are used to determine some outcome Y, and we train an algorithm to do this, we are performing supervised (machine) learning. The outcome Y may be a dependent variable (for example, the left hand side in a linear regression), or a classification (i.e., discrete outcome).

3. When we only have X variables and no separate outcome variable Y, we perform unsupervised learning. For example, cluster analysis produces groupings based on the X variables of various entities, and is a common example.

Classification Algorithms and Metrics

- Bayes Classifier: We use the **e1071** package.
- Support Vector Machines (SVM)
- Statistical Significance of the Confusion Matrix
- Word count classifiers, adjectives, and adverbs
- Fisher's discriminant
- Vector-Distance Classifier
- Accuracy
- Using the RTextTools package

Sentiment over Time

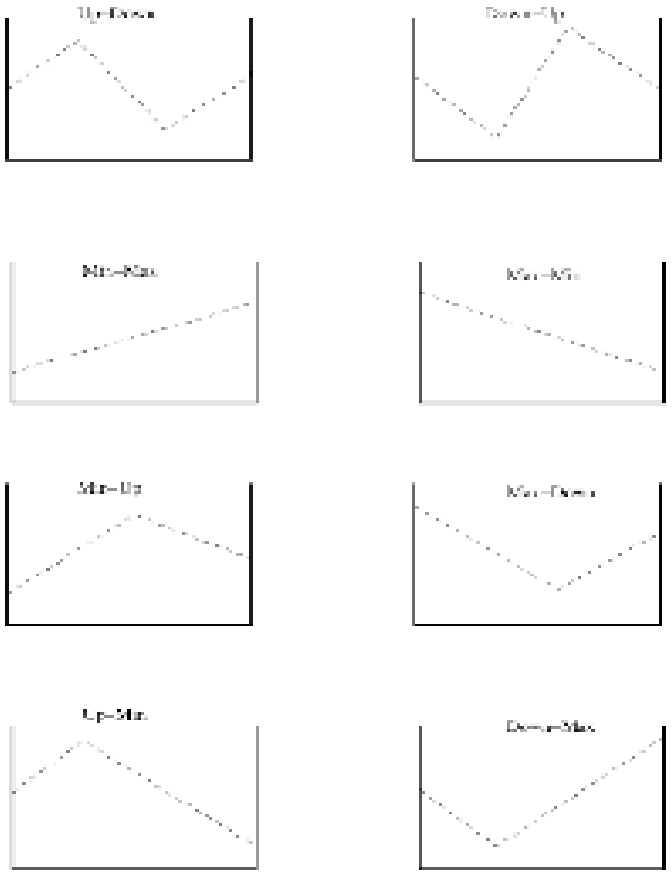


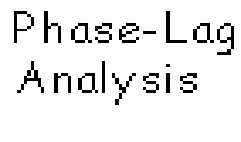
Stock Sentiment Correlations

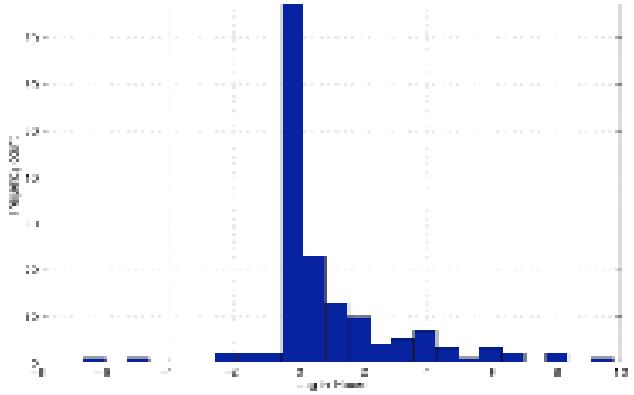
Correlations of Sentiment and Stock Returns for the MSH35 stocks and the aggregated MSH35 index. Stock returns (STKRET) are computed from close-to-close. We compute correlations using data for 88 days in the months of June, July and August 2001. Return data over the weekend is linearly interpolated, as messages continue to be posted over weekends. Daily sentiment is computed from midnight to close of trading at 4 pm (SENTY4pm).

Ticker	Correlations of SENTY4pm(t) with		
	STKRET(t)	STKRET(t+1)	STKRET(t-1)
ADP	0.086	0.138	-0.062
AMAT	-0.008	-0.049	0.067
AMZN	0.227	0.167	0.161
AOL	0.386	-0.010	0.281
BRCM	0.056	0.167	-0.007
CA	0.023	0.127	0.035
CPQ	0.260	0.161	0.239
CSCO	0.117	0.074	-0.025
DELL	0.493	-0.024	0.011
EDS	-0.017	0.000	-0.078
EMC	0.111	0.010	0.193
ERTS	0.114	-0.223	0.225
HWP	0.315	-0.097	-0.114
IBM	0.071	-0.057	0.146
INTC	0.128	-0.077	-0.007
INTU	-0.124	-0.099	-0.117
JDSU	0.126	0.056	0.047
JNPR	0.416	0.090	-0.137
LU	0.602	0.131	-0.027
MOT	-0.041	-0.014	-0.006
MSFT	0.422	0.084	0.210
MU	0.110	-0.087	0.030
NT	0.320	0.068	0.288
ORCL	0.005	0.056	-0.062
PALM	0.509	0.156	0.085
PMTC	0.080	0.005	-0.030
PSFT	0.244	-0.094	0.270
SCMR	0.240	0.197	0.060
SLR	-0.077	-0.054	-0.158
STM	-0.010	-0.062	0.161
SUNW	0.463	0.176	0.276
TLAB	0.225	0.250	0.283
TXN	0.240	-0.052	0.117
XLNX	0.261	-0.051	-0.217
YHOO	0.202	-0.038	0.222
Average correlation across 35 stocks			
	0.188	0.029	0.067
Correlation between 35 stock index and 35 stock sentiment index			
	0.486	0.178	0.288

Phase Lag Analysis







Disagreement

The metric uses the number of signed buys and sells in the day (based on a sentiment model) to determine how much difference of opinion there is in the market. The metric is computed as follows:

$$\text{DISAG} = \left|1 - \left|\frac{B - S}{B + S}\right|\right|$$

where B, S are the numbers of classified buys and sells. Note that DISAG is bounded between zero and one.

Precision and Recall

- Precision is the fraction of positives identified that are truly positive, and is also known as positive predictive value.
- Recall is the proportion of positives that are correctly identified, and is also known as sensitivity.

	Actual		
Predicted	Looking for Job	Not Looking	
Looking for Job	10	2	12
Not Looking	1	16	17
	11	18	29

In this case precision is $10/12$ and recall is $10/11$.

Readability

"Readability" is a metric of how easy it is to comprehend text.

- Gunning-Fog Index

$$0.4 \cdot \left[\frac{\#\text{words}}{\#\text{sentences}} + 100 \cdot \left(\frac{\#\text{complex words}}{\#\text{words}} \right) \right]$$

- Flesch Reading Ease Score

$$206.835 - 1.015 \left(\frac{\#\text{words}}{\#\text{sentences}} \right) - 84.6 \left(\frac{\#\text{syllables}}{\#\text{words}} \right)$$

With a range of 90-100 easily accessible by a 11-year old, 60-70 being easy to understand for 13-15 year olds, and 0-30 for university graduates.

- Flesch-Kincaid Grade Level

$$0.39 \left(\frac{\#\text{words}}{\#\text{sentences}} \right) + 11.8 \left(\frac{\#\text{syllables}}{\#\text{words}} \right) - 15.59$$

Text Summarization

It is really easy to write a summarizer in a few lines of code. The function below takes in a text array and does the needful. Each element of the array is one sentence of the document we wan summarized.

In the function we need to calculate how similar each sentence is to any other one. This could be done using cosine similarity, but here we use another approach, Jaccard similarity. Given two sentences, Jaccard similarity is the ratio of the size of the intersection word set divided by the size of the union set.

Jaccard Similarity

A document D is comprised of m sentences $s_i, i = 1, 2, \ldots, m$, where each s_i is a set of words. We compute the pairwise overlap between sentences using the **Jaccard** similarity index:

$$J_{ij} = J(s_i, s_j) = \frac{|s_i \cap s_j|}{|s_i \cup s_j|} = J_{ji}$$

The overlap is the ratio of the size of the intersect of the two word sets in sentences s_i and s_j , divided by the size of the union of the two sets. The similarity score of each sentence is computed as the row sums of the Jaccard similarity matrix.

$$S_i = \sum_{j=1}^{m} J_{ij}$$

Text Mining Research in Finance - 1

1. Lu, Chen, Chen, Hung, and Li (2010) categorize finance related textual content into three categories: (a) forums, blogs, and wikis; (b) news and research reports; and (c) content generated by firms.

2. Extracting sentiment and other information from messages posted to stock message boards such as Yahoo!, Motley Fool, Silicon Investor, Raging Bull, etc., see Tumarkin and Whitelaw (2001), Antweiler and Frank (2004), Antweiler and Frank (2005), Das, Martinez-Jerez and Tufano (2005), Das and Chen (2007).

3. Other news sources: Lexis-Nexis, Factiva, Dow Jones News, etc., see Das, Martinez-Jerez and Tufano (2005); Boudoukh, Feldman, Kogan, Richardson (2012).

Text Mining Research in Finance - 2

1. The Heard on the Street column in the Wall Street Journal has been used in work by Tetlock (2007), Tetlock, Saar-Tsechansky and Macskassay (2008); see also the use of Wall Street Journal articles by Lu, Chen, Chen, Hung, and Li (2010).

2. Thomson-Reuters NewsScope Sentiment Engine (RNSE) based on Infonics/Lexalytics algorithms and varied data on stocks and text from internal databases, see Leinweber and Sisk (2011). Zhang and Skiena (2010) develop a market neutral trading strategy using news media such as tweets, over 500 newspapers, Spinn3r RSS feeds, and LiveJournal.

Das and Chen (*Management Science* 2007)

MANAGEMENT SCIENCE
Vol. 53, No. 9, September 2007, pp. 1375–1388
ISSN 0025-1909 | EISSN 1526-5501 | 07 | 5309 | 1375

informs
DOI 10.1287/mnsc.1070.0704

Yahoo! for Amazon: Sentiment Extraction from Small Talk on the Web

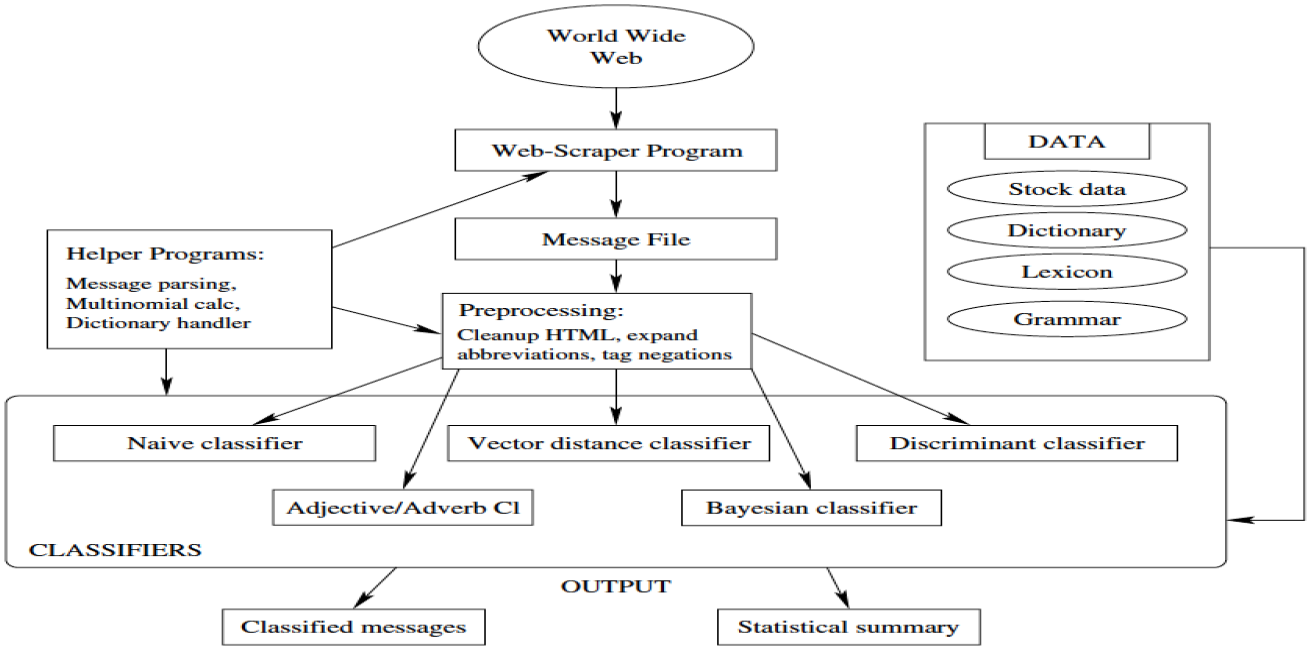
Sanjiv R. Das
Department of Finance, Leavey School of Business, Santa Clara University,
Santa Clara, California 95053, srdas@scu.edu
Mike Y. Chen
Ludic Labs, San Mateo, California 94401, mike@ludic-lab.com

Extracting sentiment from text is a hard semantic problem. We develop a methodology for extracting small investor sentiment from stock message boards. The algorithm comprises different classifier algorithms coupled together by a voting scheme. Accuracy levels are similar to widely used Bayes classifiers, but false positives are lower and sentiment accuracy higher. Time series and cross-sectional aggregation of message information improves the quality of the resultant sentiment index, particularly in the presence of slang and ambiguity. Empirical applications evidence a relationship with stock values—tech-sector postings are related to stock index levels, and to volumes and volatility. The algorithms may be used to assess the impact on investor opinion of management announcements, press releases, third-party news, and regulatory changes.

Key words: text classification; index formation; computers-computer science; artificial intelligence; finance; investment

System

Schematic of the Algorithms and System Design Used for Sentiment Extraction



Optimism Score

Message type	Optimism score	
	Mean	Std. dev.
Buy	0.032	0.075
Hold	0.026	0.069
Sell	0.016	0.071

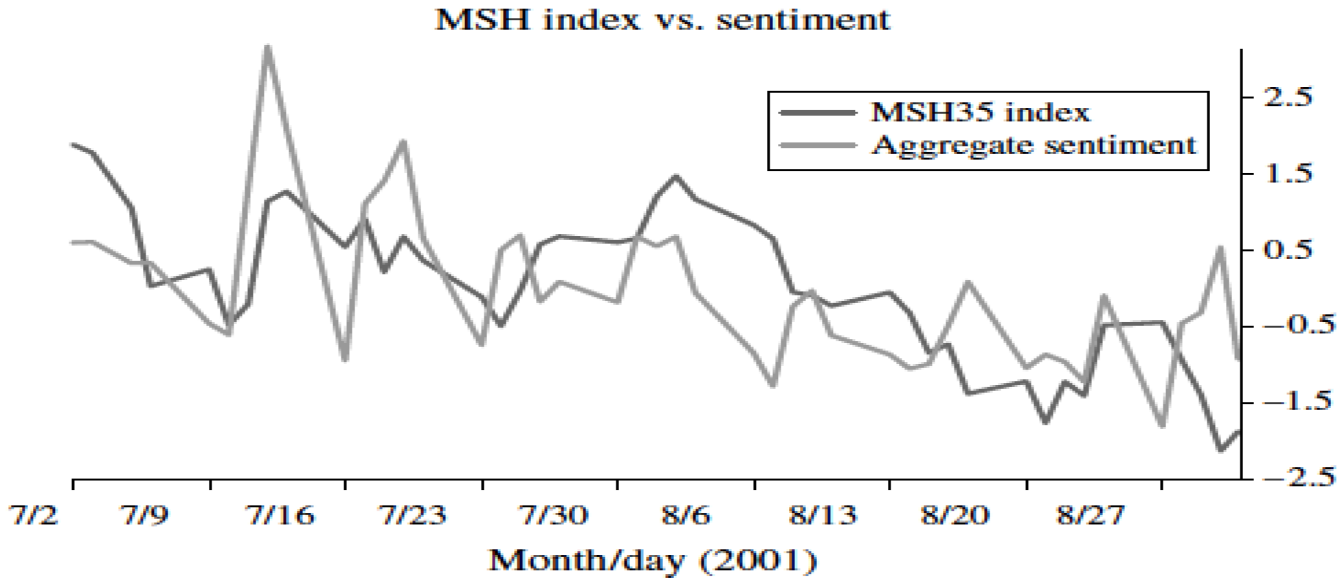
Ambiguity

Table 3 Classification as a Function of Ambiguity: Training Set Size of 913

Algorithm	Accuracy	False possitives	Sentiment error	χ^2	Number of messages
Panel A: High ambiguity					
NvWtd	45.5016	34.2224	1.8843	16.7918	7,536
vecDist	61.0934	14.2118	10.2309	236.2058	7,536
DiscWtd	54.8832	13.1900	10.0318	204.3926	7,536
AdjAdv	64.1056	33.5987	41.2818	74.0030	7,536
BayesCl	57.4443	12.3275	7.8025	238.8515	7,536
Vote3	53.3838	10.1778	14.3577	242.0414	7,536
Vote3-d	63.2705	12.1534	12.4703	227.2089	6,311
Rainbow	64.8818	32.6479	13.0191	86.8046	7,489
Panel B: Medium ambiguity					
NvWtd	46.9638	30.7494	1.0982	6.2881	1,548
vecDist	64.5995	8.6563	8.6563	69.9800	1,548
DiscWtd	58.1395	8.5917	9.4961	58.2234	1,548
AdjAdv	65.7623	28.4884	41.5375	23.2180	1,548
BayesCl	61.4341	7.7519	8.0749	67.8975	1,548
Vote3	58.3979	6.3307	13.5659	68.8180	1,548
Vote3-d	66.7671	7.3851	11.6805	65.8436	1,327
Rainbow	65.4816	28.9593	10.7304	27.4832	1,547
Panel C: Low ambiguity					
NvWtd	46.5517	25.5172	9.3103	1.9822	290
vecDist	66.8966	3.7931	7.9310	16.9034	290
DiscWtd	57.2414	5.5172	8.2759	11.7723	290
AdjAdv	61.3793	24.1379	40.0000	4.7444	290
BayesCl	64.4828	4.4828	8.2759	15.2331	290
Vote3	63.4483	4.1379	12.4138	15.3550	290
Vote3-d	66.7939	4.5802	11.0687	14.5289	262
Rainbow	67.5862	18.2759	12.0690	9.0446	290

Tech Sector Sentiment

Figure 2 **Normalized MSH Index and Aggregate Sentiment, Daily, July–August 2001**



Sentiment and Volatility

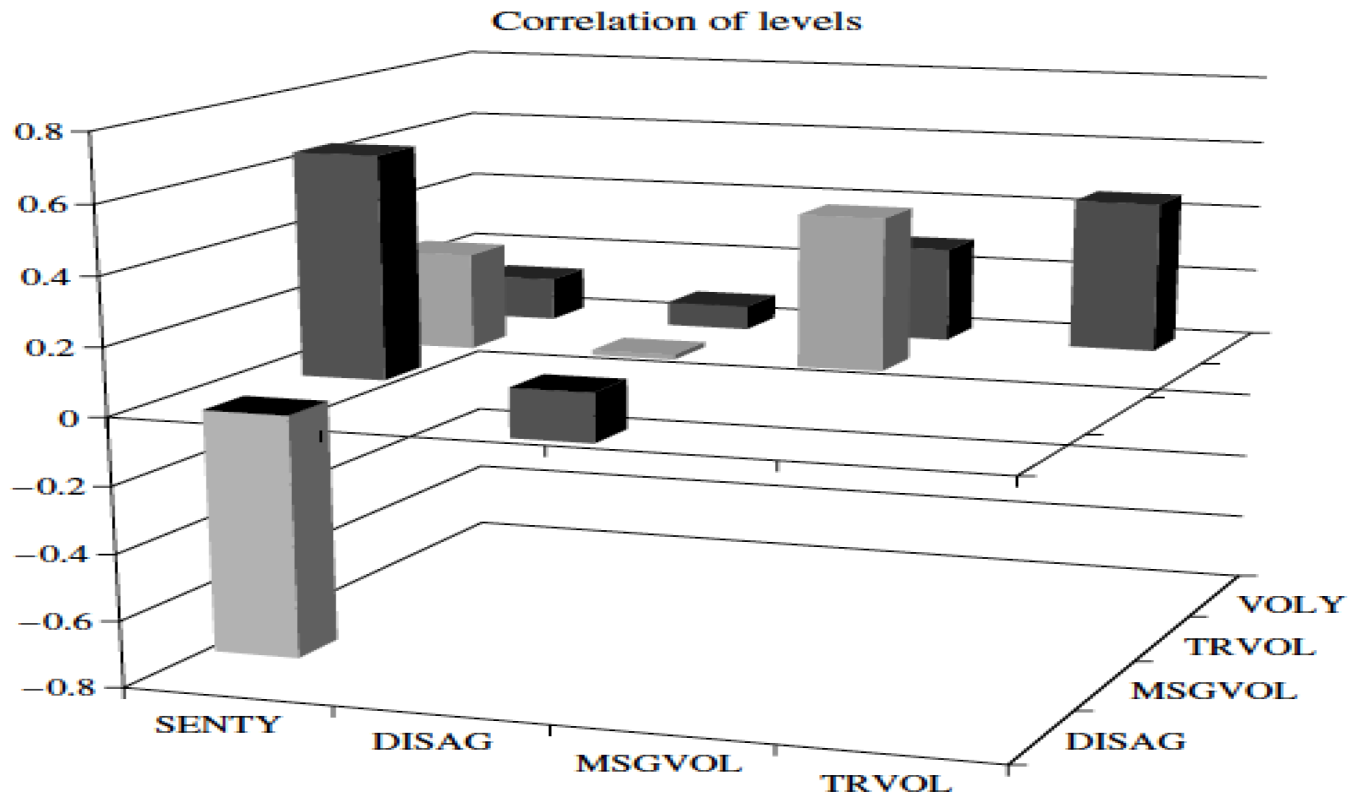
Table 5 **Relationship of Changes in Volatility and Stock Levels to Changes in Sentiment, Disagreement, Message Volume, and Trading Volume**

Independent variables	ΔVOLY	ΔSTK
Intercept	−0.000	−0.056***
	−0.01	−3.29
ΔSENTY	−0.106	0.059*
	−1.50	1.83
ΔDISAG	0.008	0.001
	0.14	0.03
ΔMSGVOL	0.197***	−0.080***
	3.34	−2.99
ΔTRVOL	0.447***	0.000
	11.84	0.01
R^2	0.20	0.02

Notes. The results relate to pooled regressions for all stocks in the sample. *t*-statistics are presented below the estimates. The normalized variables are: VOLY is volatility, STK is stock price, SENTY is the sentiment for a stock for the day, DISAG is disagreement, MSGVOL is the number of messages per day, TRVOL is the number of shares traded per day. The number of asterisks determines the level of significance: * = 10% level, ** = 5% level, *** = 1% level.

Sentiment Correlations

Figure 4 **Correlations between Sentiment, Disagreement, Message Volume, Trading Volume, and Volatility**

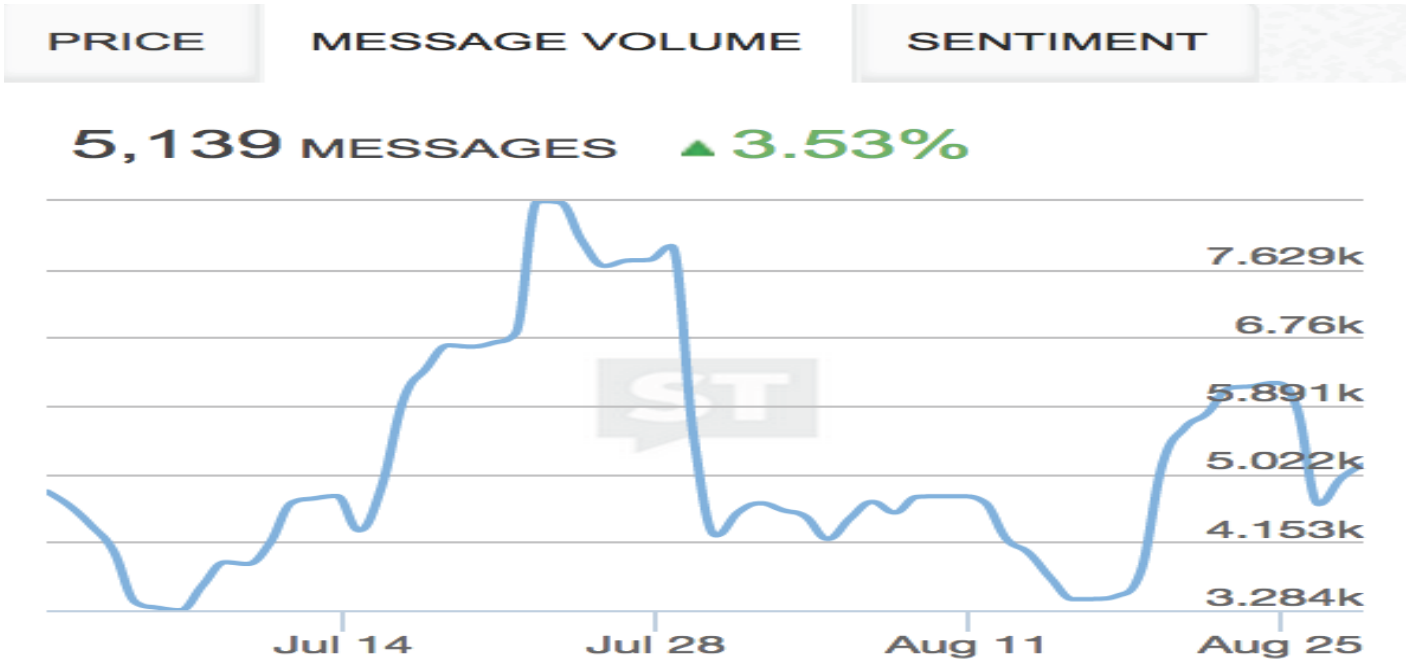


Using Twitter and Facebook for Market Prediction

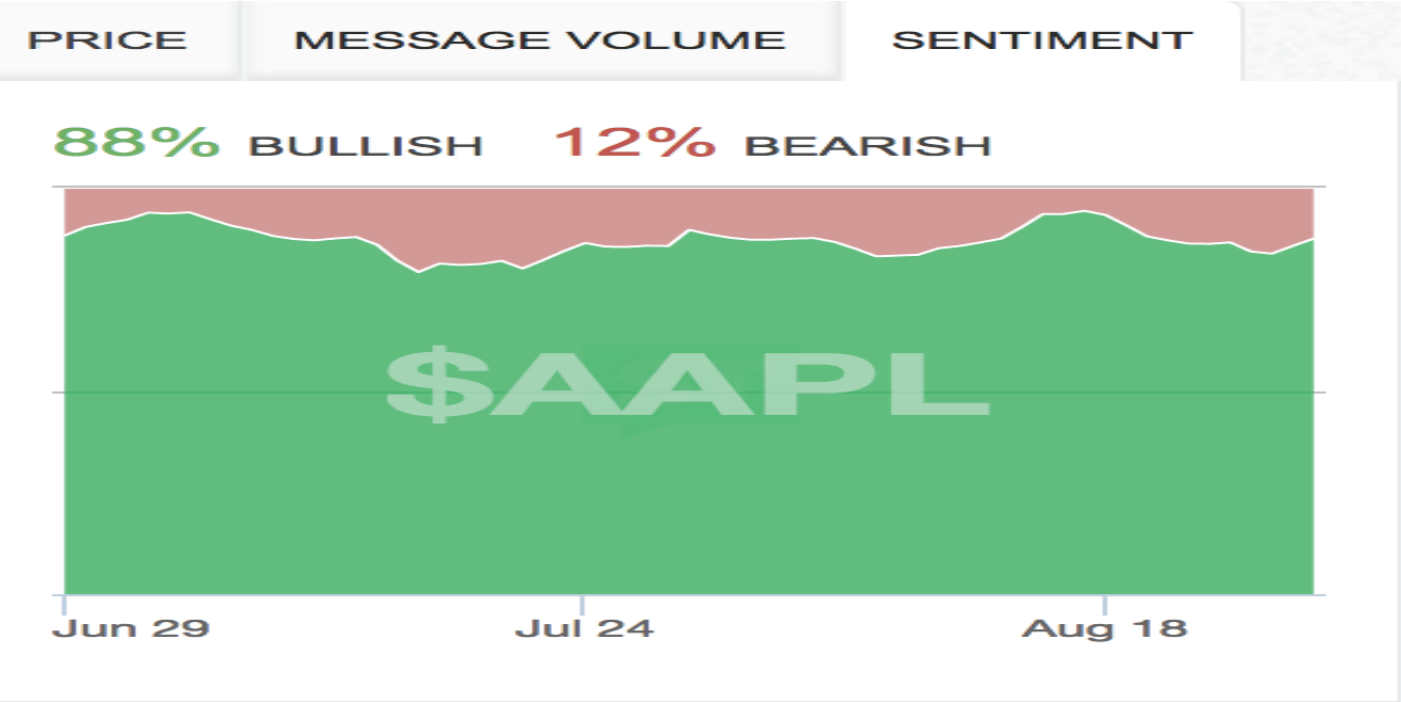
1. Bollen, Mao, and Zeng (2010): stock direction of DJIA predicted using tweets, 87.6% accuracy.
2. Bar-Haim, Dinur, Feldman, Fresko and Goldstein (2011) predict stock direction using tweets by detecting and overweighting the opinion of expert investors.
3. Brown (2012) looks at the correlation between tweets and the stock market via several measures.
4. Twitter based sentiment developed by Rao and Srivastava (2012) is correlated as high as 0.88 for returns.
5. Sprenger and Welpe (2010): tweet bullishness associated with abnormal stock returns; tweet volume predicts trading volume.

Stock Twits Prices

Messages


PRICE
MESSAGE VOLUME
SENTIMENT
5,139 MESSAGES
▲3.53%
7.629k
6.76k
5.891k
5.022k
4.153k
3.284k
Jul 14
Jul 28
Aug 11
Aug 25

Sentiment


PRICE
MESSAGE VOLUME
SENTIMENT
88% BULLISH
12% BEARISH
$AAPL
Jun 29
Jul 24
Aug 18

iSentium iSense S/T

iSentium iSense L/T

Text Mining Corporate Reports

- Text analysis is undertaken across companies in a cross-section.
- The quality of text in company reports is much better than in message postings.

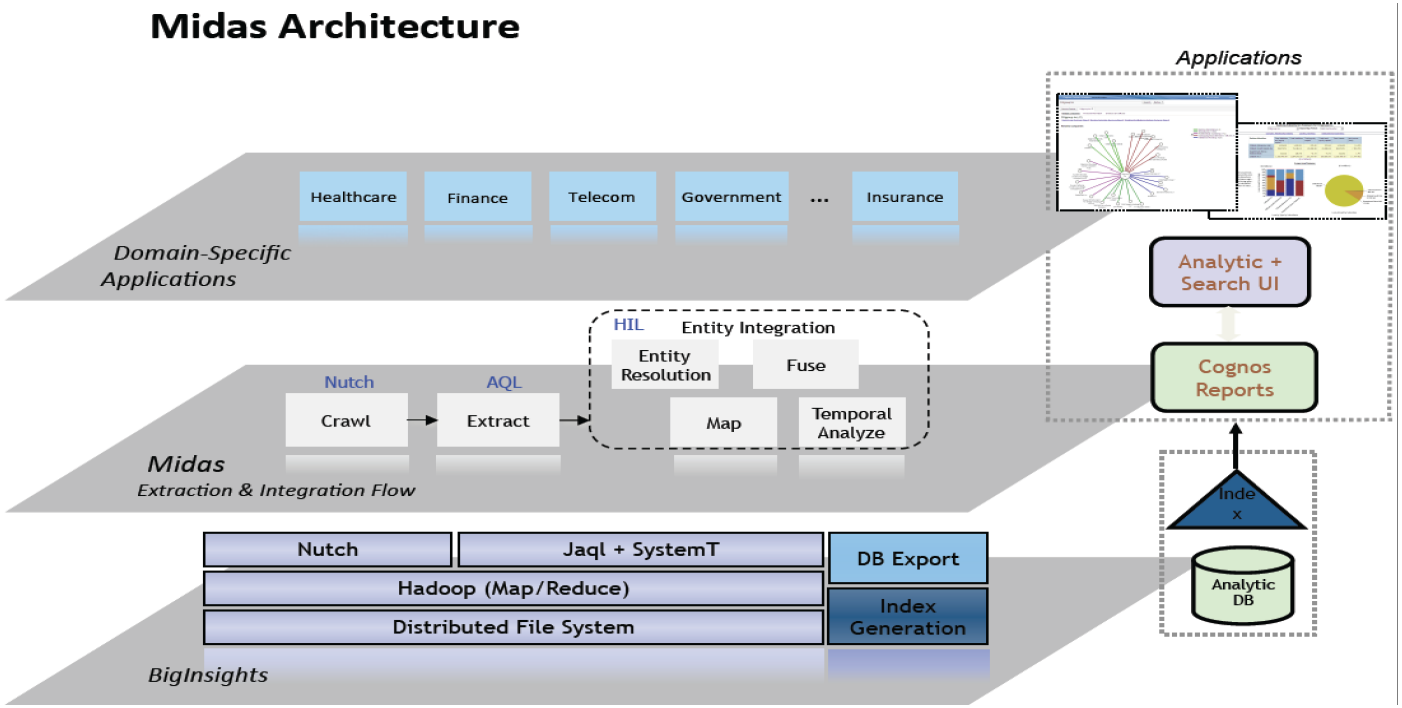
Using the MD&A

- Sharper conclusions may be possible from specific sections of the filing such as a 10-K.
- Loughran and McDonald (2011) examined whether the Management Discussion and Analysis (MD&A) section of the filing was better at providing tone (sentiment) then the entire 10-K. They found not.
- They also showed that using their six tailor-made word lists gave better results for detecting tone than did the Harvard Inquirer words.
- Proper word-weighting also improves tone detection.

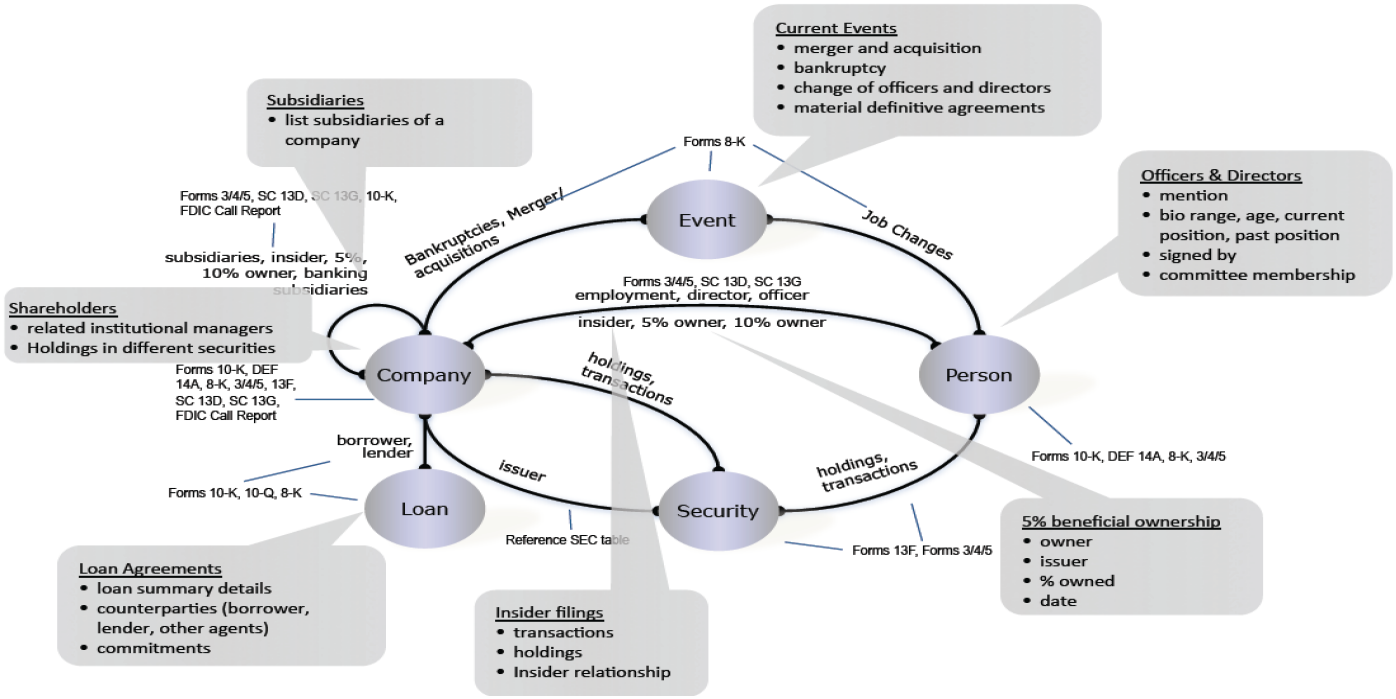
Readability of Financial Reports

- Loughran and McDonald (2014) examine the readability of financial documents, by surveying at the text in 10-K filings. They compute the Fog index for these documents and compare this to post filing measures of the information environment such as volatility of returns, dispersion of analysts recommendations.
- Fog index does not seem to correlate well with these measures of the information environment, the file size of the 10-K is a much better measure and is significantly related to return volatility, earnings forecast errors, and earnings forecast dispersion, after accounting for control variates such as size, book-to-market, lagged volatility, lagged return, and industry effects.
- Li (2008) also shows that 10-Ks with high Fog index and longer length have lower subsequent earnings.

IBM's Midas System

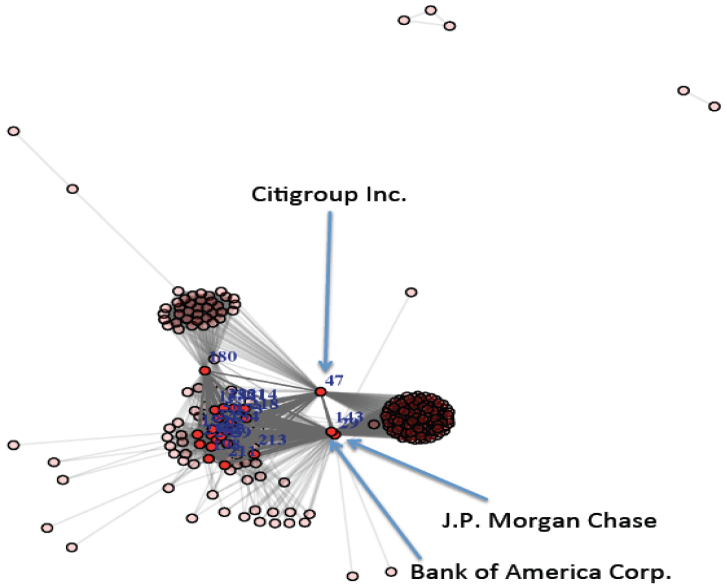


Midas Filing Map



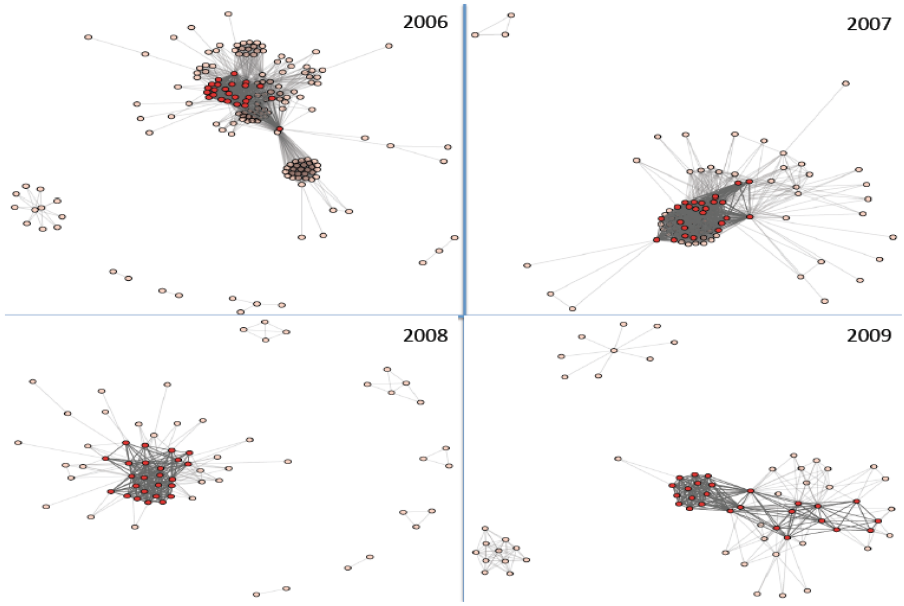
Loan Network

Loan Network 2005



Loan Network

Loan Network 2006–2009



Centrality

(Year = 2005) Financial Institution	Normalized Centrality
J P Morgan Chase & Co.	1.000
Bank of America Corp.	0.926
Citigroup Inc.	0.639
Deutsche Bank Ag New York Branch	0.636
Wachovia Bank NA	0.617
The Bank of New York	0.573
Hsbc Bank USA	0.530
Barclays Bank Plc	0.530
Keycorp	0.524
The Royal Bank of Scotland Plc	0.523
Abn Amro Bank N.V.	0.448
Merrill Lynch Bank USA	0.374
PNC Financial Services Group Inc	0.372
Morgan Stanley	0.362
Bnp Paribas	0.337
Royal Bank of Canada	0.289
The Bank of Nova Scotia	0.289
U.S. Bank NA	0.284
Calyon New York Branch	0.273
Lehman Brothers Bank Fsb	0.270
Sumitomo Mitsui Banking	0.236
Suntrust Banks Inc	0.232
UBS Loan Finance Llc	0.221
State Street Corp	0.210
Wells Fargo Bank NA	0.198

Predicting Markets

- Wysocki (1999) found that for the 50 top firms in message posting volume on Yahoo! Finance, message volume predicted next day abnormal stock returns.
- Bagnoli, Beneish, and Watts (1999) examined earnings "whispers", unofficial crowd-sourced forecasts of quarterly earnings from small investors, are more accurate than that of First Call analyst forecasts.
- Tumarkin and Whitelaw (2001) examined self-reported sentiment on the Raging Bull message board and found no predictive content, either of returns or volume.

Bullishness Index

- Antweiler and Frank (2004) bullishness index

$$B = \frac{n_B - n_S}{n_B + n_S} = \frac{R-1}{R+1} \in (-1, +1)$$

- The bullishness index does not predict returns, but returns do explain message posting. More messages are posted in periods of negative returns, but this is not a significant relationship.
- Message board postings do not predict returns.
- Disagreement (measured from postings) induces trading.
- Message posting does predict volatility at daily frequencies and intraday.
- Messages reflect public information rapidly.

Possibile Applications for Finance Firms

An illustrative list of **applications** for finance firms is as follows:

Monitoring corporate buzz. Analyzing textual data to detect, analyze, and understand the more profitable customers or products. **Targeting new clients.** Customer retention, which is a huge issue. Text mining complaints to prioritize customer remedial action makes a huge difference, especially in the insurance business. **Lending activity - automated management of profiling information for lending screening.** Market prediction and trading. **Risk management.** Automated financial analysts. **Financial forensics to prevent rogue employees from inflicting large losses.** Fraud detection. **Detecting market manipulation.** Social network analysis of clients. **Measuring institutional risk from systemic risk.**

Topic Modeling

- Latent Semantic Analysis (LSA) is an approach for reducing the dimension of the Term-Document Matrix (TDM).
- And Latent Dirichlet Allocation (LDA), what does it have to do with LSA? The **topicmodels** package.
- Latent Dirichlet Allocation (LDA) was created by David Blei, Andrew Ng, and Michael Jordan in 2003, see their paper titled "Latent Dirichlet Allocation" in the *Journal of Machine Learning Research*, pp 993–1022.

LDA as a Probability Model

The simplest way to think about LDA is as a probability model that connects documents with words and topics.

- Likelihood of the entire Corpus

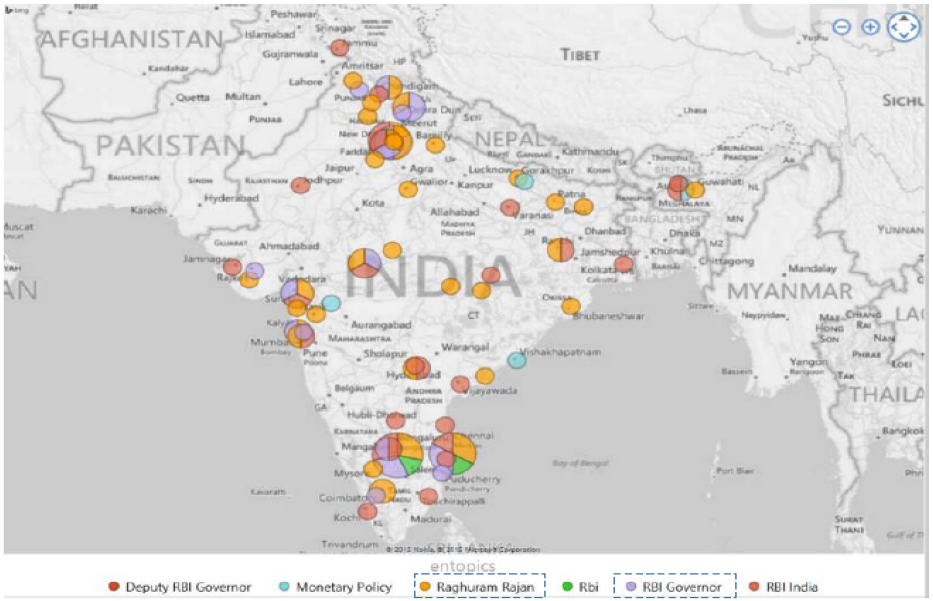
$$p(D) = \prod_{j=1}^{M} \int p(\theta_j|\alpha) \left(\prod_{l=1}^{K} \sum_{t_{jl}} p(t_l|\theta_j) p(w_l|t_l) \right) d\theta_j$$

- The goal is to maximize this likelihood by picking the vector α and the probabilities in the matrix B. (Note that were a Dirichlet distribution not used, then we could directly pick values in Matrices A and B.)

- The computation is undertaken using MCMC with Gibbs sampling as shown in the example earlier.

Examples in Finance

Conversations across India and around RBI topycs

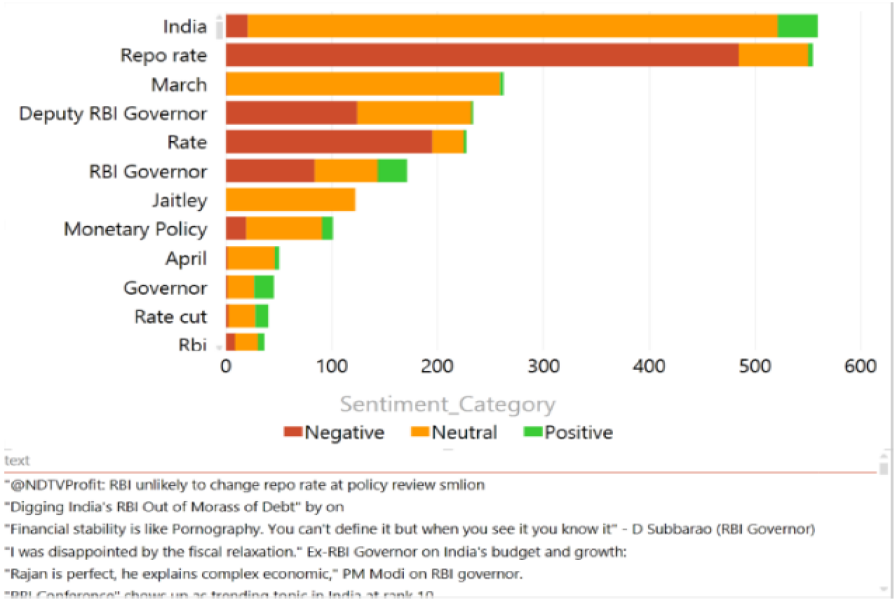


- Conversations across India on RBI, its people and the monetary policy
- Governor features in many conversations across both rural and urban areas
- Some conversations specifically around monetary policy

- Bubbles show split of conversations around Deputy RBI Governor, Monetary Policy, Raghuram Rajan, RBI and RBI Governor.
- Based on count of unique conversations
- Date Range: 1st – 14th April, 2015

Topics

Top Topics along with RBI

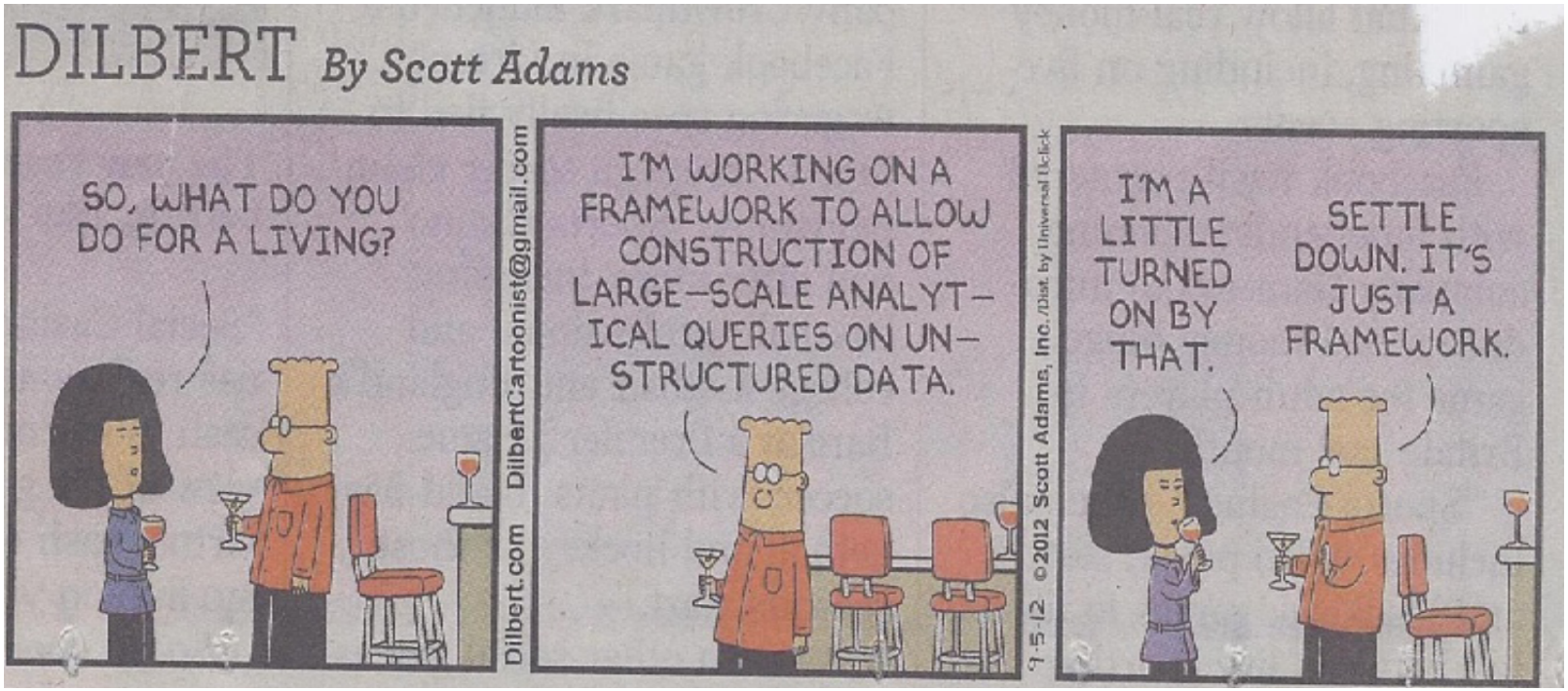


topycs

- Repo rate evokes negative sentiment as people don't expect it to be changed
- Repo rate, rate cut and monetary policy are discussed frequently with RBI

- Vertical Axis – Topics of Discussion
- Horizontal Axis – Count of Unique Conversations
- Date – 25th March - 14th of April
- Colors represent sentiment for conversation, Negative – Red, Neutral – Orange, Positive - Green

End Note!



Biblio at:
http://algo.scu.edu/~sanjivdas/Das_TextAnalyticsInFinance.pdf