Text-Driven Forecasting

Noah A. Smith*
School of Computer Science
Carnegie Mellon University
nasmith@cs.cmu.edu

March 1, 2010

Executive Summary

Forecasting the future hinges on understanding the present. The web—particularly the *social* web—now gives us an up-to-the-minute snapshot of the world as it is and as it is perceived by many people, *right now*, but that snapshot is distributed in a way that is incomprehensible to a human. Much of this data is encoded in text, which is noisy, unstructured, and sparse; yet recent developments in natural language processing now permit us to analyze text and connect it to real-world measurable phenomena through statistical models. We propose **text-driven forecasting** as a challenge for natural language processing and machine learning:

> Given a body of text T pertinent to a social phenomenon, make a concrete prediction about a measurement M of that phenomenon, obtainable only in the future, that rivals the best-known methods for forecasting M.

We seek methods that work in many settings, for many kinds of text and many kinds of measurements.

Accurate text-driven forecasting will be of use to the intelligence community, policymakers, and businesses. The use of statistical models is the norm of natural language processing methods, making it straightforward to develop models that provide posterior probabilities over measurements. Evaluation and comparison of forecasting algorithms is straightforward and inexpensive. We present encouraging recent results across several domains, emphasizing that a broad suite of forecasting problems and text sources will best support progress on this task.

Further, advances in text-driven forecasting will have broad impact in natural language processing, giving a concrete, theory-independent platform that encourages exploration of new ideas for tackling various aspects of text-oriented computational intelligence.

*The views in this paper are my own, but they were strongly influenced through collaboration with Bryan Routledge (Tepper School of Business, CMU), Shimon Kogan (U. Texas), Jacob Sagi (Vanderbilt U.), and William Cohen (CMU). I am indebted to participants in my Fall 2009 seminar at CMU, "Text-Driven Forecasting" (11-773), particularly Brendan O'Connor and Tae Yano, for stimulating discussions.

1 Introduction

The rise of the web as a medium for social interaction and social expression has generated much interest across the scientific disciplines. The *social* web offers a new window—that is, a new source of data—revealing the ways humans interact and communicate. The unprecedentedly large "crowd" of social web inhabitants may be a source of collective wisdom that supercedes or complements more traditional sources of knowledge.

The social web puts into sharp relief all of the classic challenges of computational intelligence, notably the challenge of **language understanding**. At best, text is difficult to treat automatically: people speak in diverse languages and dialects, each with its own massive, sparse vocabulary that changes daily and its own particular grammar rules and exceptions. Most sentences are ambiguous and require interpretation in non-literal ways to be meaningful. Text on the social web is even harder for computational systems to cope with. Social text is as often about opinions, perceptions, and emotions as it is about facts. If we cannot computationally represent the meaning of a news story about an earthquake, how can we hope to represent the meaning of a barrage of microblog posts instantly following that earthquake?

Although the challenges of language processing for the social web are daunting, the social web brings with it a remarkably novel asset: the text is tied in to a vast structure of metadata, including widely-studied hyperlink structure, but also temporally linked metadata. Such metadata serve to "ground" the text in a concrete, quantitative, "$\mathbb{R}$" world. This connection is only available now, as the two kinds of data—real-time measurements and real-time text that discusses the same phenomena being measured—are finally brought together by the social web in myriad instantiations.

We propose the following challenge: **Given a body of text T pertinent to a social phenomenon, make a concrete prediction about a measurement M of that phenomenon, obtainable only in the future, that rivals the best-known methods for forecasting M.** We conjecture that solutions to this problem will have direct utility in a wide range of intelligent computing applications, and that a research program devoted to this general challenge will shed light on long-standing problems in natural language processing.

The remainder of this document highlights some recent successes with text-driven forecasting using natural language processing methods. We emphasize that each of these tasks provides a test bed that is *neutral* with respect to linguistic theories. Any model that relates noisy text to measurable, quantitative real-world phenomena can compete. If the model is probabilistic, posterior probabilities over measurements are straightforwardly calculated. Objective evaluation is cheap and straightforward: what is the error in the forecast $\hat{M}$?

2 Movie Reviews and Gross Revenues

Forecasting involving profit has obvious applications and tends to generate interest. Before a film hits the box office, critics attend advance viewings and publish reviews about it. Some researchers have constructed computational models that analyze the *sentiment* in a review: is the critic positively or negatively inclined toward the film (Pang et al., 2002)? A more concrete question is this: how much will the film earn at the box office? This prediction problem has been considered using many features of a film—its rating, its genre, whether major stars appear in it, its budget—but here we consider making the prediction using text produced by expert critics.

We consider 1,351 movies released between January 2005 and June 2009. For each movie, we

Model	Mean Absolute Error ($)
Baseline Predict median from training data	$7,097
Metadata (D) {U.S. origin?, log budget, # screens, runtime}	7,313
Text (T) Words, bigrams, trigrams, and dependency relations	6,729
Metadata and text together (D, T)	6,725

Table 1: Predicting per-screen gross revenues of movies. Error is calculated on a test set of 180 movies. Text reduces error by 8% compared to metadata and 5% against the strong baseline of predicting the training set median.

obtained two kinds of data The first is metadata from `www.metacritic.com`: name, production house, genre(s), scriptwriter(s), director(s), country of origin, primary actors, release date, MPAA rating, and running time; and from `www.the-numbers.com`: production budget, opening weekend gross revenue, and number of screens on which it played in that weekend. The second is reviews drawn from six review websites that appeared most frequently at `www.metacritic.com`. Only reviews from *before* the release date of the movie were collected, ensuring that no information from the opening weekend could appear in or influence the review.

An instance consists of the a movie's metadata (D), its reviews (T), and its per-screen gross revenue during the opening weekend (M). The goal is to build a forecaster of M, using T, that outperforms a forecaster based only on D (as in past work). We applied linear regression modeling with state-of-the-art "elastic net" regularization (Zou and Hastie, 2005; Friedman et al., 2008):

$$\min_{\boldsymbol{w}} \frac{1}{n} \sum_{i=1}^{n} (M_i - \boldsymbol{w}^\top \underbrace{\boldsymbol{f}_i}_{\text{depends on } D_i, T_i, \text{ or both}})^2 + \lambda_1 \|\boldsymbol{w}\|_1 + \lambda_2 \|\boldsymbol{w}\|_2^2 \quad (1)$$

where n training examples are used for learning. The model is trained on 988 examples from 2005–2007 and regularization constants λ_1 and λ_2 are tuned on examples from January–August 2008. We evaluate the model by forecasting M for each movie in September 2008–June 2009. We calculate the mean absolute error (MAE) between $\hat{M}$ and the actual gross earnings per screen M. Objectively measured results are striking; see Table 1. Models that use text alone or in addition to metadata outperform the metadata model. (More variations of this formulation and experiments are explored in a forthcoming paper, Joshi et al., 2010.) Note that our regression model results in sparse coefficients (most $w_i = 0$). The metadata-only model selected only four features (see Table 1). The combined model ruled out all but two non-text features (log budget and U.S. origin), effectively deciding that the text contained most of the information in the metadata. This is consistent with an inspection of the models; see Joshi et al. (2010), where we see strongly-weighted words corresponding to ratings, genres, specific people, as well as sentiment words. This model is *probabilistic* and can be used to calculate posterior distributions over earnings.

3 SEC Reports and Volatility of Returns

It is widely known that predicting returns or profit is a difficult problem. There are, however, other properties of financial markets that may be predicted, and which are still useful. **Risk** is one such example; in Finance, risk is often quantified in terms of volatility, or the standard deviation of returns over a period of time. In financial reports mandated by the Securities Exchange Commission, publicly traded U.S. firms are required to disclose (among other things) sources of risk. We consider annual reports known as "Form 10-K" and aim to predict volatility in the year following a report's publication.

Model	Mean Squared Error of $\log \hat{V}^{(+12)}$
Baseline: $\hat{V}^{(+12)} = V^{(-12)}$	0.1873
Text (T) only: Words and bigrams	0.1729
History and text together ($V^{(-12)}, T$)	0.1599

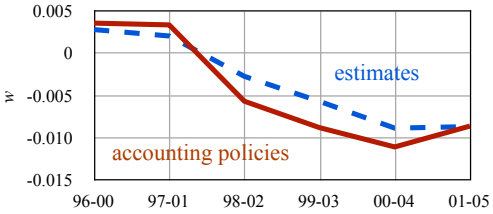

Figure 1: Left: Predicting volatility for $n = 3{,}612$ firms following their 2003 10-K reports (training on 12,853 reports from 1998–2002). The non-text baseline is historical volatility, $V^{(-12)}$. Right: Change in the regression coefficients for two words, over time. *Accounting policies* and *estimates* were weak indicators of high log-volatility, but later became strong indicators of *low* log-volatility.

"High Volatility" Terms:	*loss, net loss, expenses, year #, obligations, financing, convertible, additional*
"Low Volatility" Terms:	*rate, net income, properties, income, dividends, distributions, earnings, unsecured*

Table 2: Terms with strong regression coefficients in the model for 1998–2002. # matches any numeral.

We collected 26,806 10-K reports, consisting of a quarter billion words, from 1996–2006, and used financial data to calculate the volatility for the firm that published each report in two periods: the twelve months before the report ($V^{(-12)}$) and the twelve months after ($V^{(+12)}$). The latter is the target value we seek to predict. Because volatility shows strong autocorrelation, simply predicting $V^{(+12)} = V^{(-12)}$ performs as well as many existing financial models.

We used a linear regression model, in this case trained using support vector regression (Drucker et al., 1997), to predict $V^{(+12)}$ from word and bigram frequencies in Section 7 of 10-K reports, optionally including $V^{(-12)}$ as a feature. Figure 1 (left) shows one set of experimental results where the text-only model outperforms the baseline, and where the two together work even better. Table 2 shows features that were learned to be strong indicators of high and low volatility. Many more experimental results are presented in our recent paper, Kogan et al. (2009). (E.g., as a result of constructing models of volatility from text, we found that reports from after 2002 were more informative about volatility (lower error). Further analysis has led to a strong case that the Sarbanes-Oxley Act of 2002, passed by the U.S. Congress to reform financial reporting, and related reforms, affected 10-K informativeness.)

4 Political Blog Reader Behavior

So far we have discussed scenarios where the text provides information about the future; our interest in the text ceases once future events occur. A different situation is one where the text is intrinsically interesting, and we wish to measure that interest. Can we use properties of a piece of text to predict its inherent interest to potential readers?

For example, we created a dataset of 7,930 blog posts drawn from five American political blogs in 2007–8.[1] Each blog allows readers to leave comments on the post, with comments marked by a site-specific username. In two studies, we have predicted the individual (Yano et al., 2009) and aggregate (Yano and Smith, 2010) behavior of blog readers.

[1] Our dataset is publicly available at `http://www.ark.cs.cmu.edu/blog-data`.

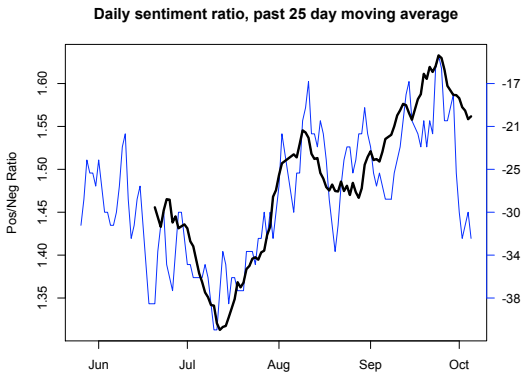

Figure 2: The lighter, blue, more volatile line shows the Gallup daily "Economic Confidence"index during June–October 2009. The darker black line is a smoothed ratio of positive to negative words per day, using the sentiment lexicon of Wilson et al. (2005) and past-25-day smoothing. The ratio is calculated from Twitter "tweets" that mention the word *economy*.

For this problem we experimented with hidden-variable models based on latent Dirichlet allocation (LDA; Blei et al., 2003). Learning such models produces not just a forecaster, but "topics," or soft word clusters that tend to be topically coherent and can be quantitatively linked to the prediction. Here we summarize some key findings relating to the single-author, left-leaning blog `http://www.matthewyglesias.theatlantic.com`; see the papers cited above for much more. For the task of predicting the five most likely commenters per new post (in unseen test data), our "CommentLDA" model achieved 27.5% precision, compared to a Naïve Bayes "bag of words" baseline achieving 25.1%. More strikingly, the model discovered topics relating to religion, domestic policy, the Iraq war, energy, and the Democratic primary election, among others. Qualitatively, commenters tend to be more off-topic, more direct, more emotional, and talk about more tangible things than the blog author. For the task of predicting which posts would achieve more than average comments (measured in the number of words), we built a model combining CommentLDA with Poisson regression. Compared to Naïve Bayes "bag of words," our precision drops slightly (72.5% to 70.2%), but recall increases from 41.7% to 68.8%. The model learned, for example, that race, civil rights, and policy discussions get readers more excited than the blogger's musings on sports and book reviews.

Blogs are complex, coupling language with social systems. Unlike widely-studied news, the language does not aim to merely report facts, but to persuade, motivate, make fun, and argue. We believe the blogosphere offers tremendous opportunities for forecasting applications; our work so far has only focused on single-blog, blog-internal predictions.

5 Microblogs and Public Opinion

An even more dynamic, broad-participation platform is exemplified by microblogs like Twitter. Twitter users post short (average eleven words) updates at short intervals. Any one "tweet" is unlikely to be informative, but taken as a whole, we have found remarkable trends. Figure 2 shows how a simple aggregate score based on positive- and negative-sentiment word frequencies closely tracks a time series of tremendous interest to investors. The score is derived from tweets mentioning the word *economy*. The time series is Gallup's "Economic Confidence" index, which is reported daily based on three days of polling.[2]

Polling is expensive and slow; it requires asking many people what they think. If public opinion can be measured from free, openly available data that people generate *without* being prompted, then

[2] `http://www.gallup.com/poll/122840/gallup-daily-economic-indexes.aspx`

we may also be able to predict trends more quickly than using traditional polling methods. A more full discussion is presented in O'Connor et al. (2010); we have also found promising connections between Twitter text and presidential approval ratings.

6 Additional Scenarios

Discussing the above results has led to a huge number of ideas about where else text-driven forecasting might be useful. A few follow; many more are conceivable.

"Text" need not refer to assertions posted for the world to read. For example, Google researchers have found that search queries submitted to its search engine—a variation on text—can predict trends in the spread of contagious disease (Ginsberg et al., 2009).

The language of policymaking lawmaking is often inscrutable. By building forecasting models that predict lawmakers' votes on a bill, we may discover systematic connections between political positions and coalitions and this domain's language (Thomas et al., 2006).

None of the techniques used so far rely heavily on language-specific resources.[3] Hence text-driven forecasters can be built for text in any language that can be segmented. We expect that deeper linguistic processing will improve these models, but an intriguing idea is to use the information discovered by the forecasting model to support the development of language processing resources and tools for additional languages.

7 Challenges

The obvious challenges are (i) selecting the right text data to answer a given forecasting question, (ii) appropriately analyzing those data, (iii) discovering the complex relationship between textual clues and future measurements through sophisticated but scalable machine learning, and (iv) building techniques that are applicable across many forecasting problems. Some short-term methodological issues involve appropriate evaluations when the data come in a time series (as in §5) and hence are not IID, and defining the characteristics of a problem where text-driven forecasting has the potential to be useful and accurate (i.e., information about the future *is present* in the text, if hard to extract). Other challenges that could develop into long-term research plans include improving the interpretability of models (not just *what* is predicted, but *why*?), improving the generality of the technique, so that a broad range of predictions are possible, reporting predictions with confidence scores, and learning what information is trustworthy.

8 Conclusion

Text-driven forecasting holds promise for advancing the state of the art in intelligent text processing that is grounded in objective reality. The framework is fundamentally empirical, will inspire young scientists, and is amenable to a huge diversity of approaches building on different linguistic and computational theories. The problem is hard—many future events cannot be forecasted using text—but recent advances in natural language processing and machine learning suggest that the time is right to address the challenges.

[3] The parsing used in §2 can be removed with a slight performance degradation, or an unsupervised parser can be trained for a new language; see Smith, 2006.

References

D. M. Blei, A. Ng, and M. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

H. Drucker, C. J. C. Burges, L. Kaufman, A. Smola, and V. Vapnik. Support vector regression machines. In *Advances in Neural Information Processing Systems 9*, 1997.

J. Friedman, T. Hastie, and R. Tibshirani. Regularized paths for generalized linear models via coordinate descent. Technical report, Stanford University, 2008.

J. Ginsberg, M. H. Mohebbi, R. S. Patel, L. Brammer, M. S. Smolinski, and L. Brilliant. Detecting influenza epidemics using search engine query data. *Nature*, 457:1012–1014, 2009.

M. Joshi, D. Das, K. Gimpel, and N. A. Smith. Movie reviews and revenues: An experiment in text regression. In *Proceedings of Human Language Technologies Conference of the North American Chapter of the Association for Computational Linguistics*, 2010.

S. Kogan, D. Levin, B. R. Routledge, J. S. Sagi, and N. A. Smith. Predicting risk from financial reports with regression. In *Proceedings of the Human Language Technologies Conference of the North American Chapter of the Association for Computational Linguistics*, 2009.

B. O'Connor, R. Balasubramanyan, B. R. Routledge, and N. A. Smith. From tweets to polls: Linking text sentiment to public opinion time series. 2010. In review.

B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2002.

N. A. Smith. *Novel Estimation Methods for Unsupervised Discovery of Latent Structure in Natural Language Text*. PhD thesis, Department of Computer Science, Johns Hopkins University, 2006.

M. Thomas, B. Pang, and L. Lee. Get out the vote: Determining support or opposition from Congressional floor-debate transcripts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2006.

T. Wilson, P. Hoffmann, S. Somasundaran, J. Kessler, J. Wiebe, Y. Choi, C. Cardie, E. Riloff, and S. Patwardhan. Opinionfinder: A system for subjectivity analysis. In *Proceedings of Human Language Technologies-Empirical Methods in Natural Language Processing Conference Interactive Demonstrations*, 2005.

T. Yano and N. A. Smith. What's worthy of comment? content and comment volume in political blogs. 2010. In review.

T. Yano, W. W. Cohen, and N. A. Smith. Predicting response to political blog posts with topic models. In *Proceedings of Human Language Technologies Conference of the North American Chapter of the Association for Computational Linguistics*, 2009.

H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal Of The Royal Statistical Society Series B*, 67(5), 2005.