INTERNATIONAL INSTITUTE FOR ANALYTICS

Enterprise Research Service

Research Brief

Extracting Value from Unstructured Text

June 2014

Authored by:

Andrew Fast

John F. Elder, IV

Key Takeaways

1. Natural language contained in unstructured text presents a significant challenge for automated analytical approaches due to the wide variety in forms of expression including technical language, sarcasm, and colloquialisms.
2. The information contained in unstructured text is highly valuable when considered alone, and the value multiplies when the text can be associated with available structured data such as case outcomes or numerical measures of performance.
3. Successful strategies for extracting value from text always include processing at two levels: first, at the word level to identify key concepts and second, at the document level to associate the collection of concepts contained within a document with a specific business outcome.

An Obvious Need

Many estimates claim that over 80% of the world's data is stored as unstructured text. Whatever the exact proportion, there is no denying that a significant amount of valuable information is stored within free-text documents such as reports, memorandums, and correspondence. Text is also stored as "notes" or "free-text" fields within an otherwise structured database. While some of the results presented in a report may also be available as structured data, the most important insights – such as expert opinions or final conclusions - are almost always only available as free text. This also holds for call center notes, electronic medical records, and claims notes.

Even for an analytics professional well equipped to handle traditional structured data (numerical and categorical data) the challenge is that text is different and does not work well with standard analytics tools and expertise. This mismatch is responsible, in part, for Gartner placing Text Analytics in the "Trough of Disillusionment" in their Hype Cycle for Big Data (July 2013). Despite this unpromising judgment, it is possible to successfully extract value from unstructured text. In this research brief, we share our lessons learned from use cases across multiple industries, and describe a successful case study using call center text to improve a model of churn propensity for a mobile phone provider.

What Makes Text So Difficult?

In order to capture the breadth of human experience, written communication has evolved into a wide range of textual expression including everything from long philosophical treatises to a 140-character tweet. When compared with structured data, the task of understanding even the shortest tweet (e.g., "*LOL*") can be quite complex. A numerical or categorical value stored in a database stands alone and requires little or no context to be interpreted accurately. In contrast, text requires background knowledge of the domain and an understanding of any specialized language or technical jargon used by participants. Even with the proper domain knowledge and understanding, stylized writing including humor and sarcasm is often interpreted differently by different people. This complexity precludes a simple reshaping of text into a structured form for a large corpus of documents.

If there were only a modest number of text documents to process, it would be practical for humans to read and categorize each one. Most document corpora, however, are much too large to process manually, and yet much too complex for traditional analytic software to handle well. It is not surprising then, that the first new algorithms for "Big Data", such as Google's MapReduce framework, were designed to handle large amounts of text. The vast quantity of data also increases the amount of useless and irrelevant text within the corpus. And, though the text may contain useful information, it will not be valuable unless it can be associated with a specific business problem. This provides the context by which we can make judgments about particular terms and use cases.

Finally, each textual document must be processed twice to extract all of the value. First, the individual words must be grouped into phrases and concepts. This natural language processing (NLP) or concept extraction, condenses multiple variations of words into a single form, simplifying future work. To do this initial processing well requires an understanding of the linguistic structures of the text. Once the words have been processed, the second step is to process the entire document to associate the NLP features with the specific business outcomes of interest.

In summary, dealing with textual data requires a different mindset and much more work than with structured data.

Opportunities and Value

Experience across a variety of application areas has shown that incorporating text has always increased the value of analytics projects as the text provides deeper insights about the true

cause of behavior than either numerical or categorical data. Similarly, structured data enriches the value of text by providing unambiguous context for its correct interpretation. Incorporating text into our classification models has consistently improved model accuracy.

No matter which classification algorithm - such as a random forest or neural networks - is employed, providing access to text information can be very useful for analysts because the text contains human-entered content beyond what is available in the structured data. It is a powerful practice to create a searchable text database or search engine to provide access to the text. Searching text provides some immediate value with very little effort, often unlocking rich data within text fields that are not otherwise included in company-wide knowledge management efforts. Likewise, employing structured data into search (known as using *search facets*) enhances the value of the search-returned data and any subsequent classification model.

We have found that combining categorical, numerical, and textual data (when available) into a single predictive model is the best way to achieve maximum performance. Next, we detail a successful modeling effort using text from call notes to help predict whether someone is likely to leave their mobile phone provider. The case study is followed by some detailed technical recommendations for extracting value from text and advice for how to achieve success with your first text project.

Case Study: Improving Customer Retention and Profitability for a Regional Provider of Wireless Services

A compelling example application showing the value of combining textual data with related structural data comes from a mobile phone carrier seeking to limit customer “churn”. Competition in the wireless telecommunications industry is fierce, with providers fighting for subscribers. Churn occurs when a carrier loses a subscriber, because the customer fails to renew an expiring contract, fails to pay their monthly bill, or terminates their contract early.

This regional carrier provides wireless communications services to consumers and businesses in seven states with approximately 400,000 subscribers. They cannot tolerate many subscriber losses without impacting profitability. In the year prior to the implementation of the strategy, churn became a significant problem, and the carrier turned to analytics to help stem the losses. The strategy of combining structured data with text allowed the carrier to reduce churn by 17 percent in the first year of operation. Since then, churn has dropped to its lowest point in years, the subscriber base is growing, and the analytics effort has paid for itself many times over; in fact, it has repaid its entire cost *each month* it has been in use!

The Analytics Approach

In this case, churn was treated as a supervised classification problem where the target label indicated whether the subscriber would leave within 90 days. Structured data and unstructured text from the call center notes were used as inputs into the models. The structured data contained information such as the type of contract, the length of time remaining on the contract, the customer's credit score, the type of plan, minutes used, etc. These structured variables are highly predictive but can only suggest a few actions to the marketing department to prevent likely churners from leaving the network.

Processing Call Notes

To help understand reasons for churn and improve overall model accuracy, data from the call center notes were added as inputs. These notes contained a record of all customer service interaction with the subscriber such as complaints and offers, as well as a collection of technical information such as the model of phone mentioned in the interaction. The call notes were also the most reliable source of phone information such as breakage or device transfers.

In order to take full advantage of the call notes, word-level processing was performed to identify phrases and to group similar semantic concepts together. After strings of characters were combined into “tokens”, linguistic rules were used to identify words and phrases indicating overall levels of (dis)satisfaction and to unify different mentions of wireless telecommunications jargon including terms for network coverage, cost, dropped calls, etc. To extract phone models, a lexicon of phone model names approved for use on the network was applied. Popular brand names of phones such as Samsung, LG, Blackberry, and as a special case “iPhone” were then added to the list. At the time of this analysis, the carrier had not yet received permission to sell the iPhone so the suspicion was that anyone who mentioned the iPhone by name was likely to churn.

Next, the textual features extracted in the word level processing must be incorporated into a record-level model to associate the features with the churn result. The first way to do this, known as the "Bag of Words," merges the textual data with the structured data by creating a single numeric feature for each word. The value of this feature can be a binary indicator value, an integer count, or a weight such as term-frequency, inverse document frequency (tf-idf). A second approach for incorporating textual features creates only a single text-data score and not a feature for every word. In this application, the second approach was followed, building a classification model using textual features and introducing its final prediction as a single new feature alongside the structured data for the final churn model to potentially use.

Classifying Text with a Naïve Bayes Classifier

To build a stand-alone classification model from the textual features, a variation of the "Bag of Words" approach was used, where each phrase, concept, and phone model were used as an input feature rather than each individual word. The target variable, churn rate within 90 days, was the same as for the structured data model. Choosing the same target label helps (but is not necessary) for integrating the two models later.

A Naïve Bayes classifier with empirical Bayesian initialization was then trained. A Naïve Bayes classifier is a popular choice for text because it is very efficient to train, even on datasets that contain a large number of rows and columns. During model training, each feature is assigned a weight based on the number of times the feature co-occurs with each outcome (churn vs. no churn). For classification, each customer receives an overall score computed from the weights of each of the phrases and concepts appearing in that customer's notes. Individual feature weights can also be used to identify phrases and concepts that are highly correlated with likelihood to churn.

Textual datasets can sometimes be problematic for traditional analytics software due to the high dimensionality caused by creating a new feature for each word. This high dimensionality also can lead to problems with data sparsity. A phrase or concept that appears only a few times may appear to be disproportionately correlated with churn due to the small sample size. To counter this effect, non-zero initialization of the Naïve Bayes model weights should be used. Rather than initializing the counts at zero, assigning a small non-zero count to each feature that is tied to the overall propensity of churn is a better approach. Given sufficient evidence, this adjustment becomes inconsequential in the final decision. But with sparse data, the initial count helps prevent overfitting the model on rare features. This approach was implemented using a custom script within a major commercial analytics platform.

Phone Models Highly Correlated With Churn

One of the major motivations for incorporating text is to improve the actionability of the gathered insights. Using the weights from the Naïve Bayes classifier, a series of specific phone models that were highly correlated with churn were identified. This led to new offers to encourage subscribers with those phones to upgrade to a more modern device. The carrier already had a well-defined process for identifying and addressing subscribers who were complaining about the cost of service. These customers were also easy to identify and address. Identifying problematic phone models helped advance the carrier's marketing efforts.

A word cloud visualization of the different phone models is shown in Figure 1. There, the size of the words is tied to the frequency with which they occur. More frequent words are larger, and less common words are smaller. Furthermore, words are colored based on their correlation

with churn. Red indicates phone models highly correlated with churn, and blue low. Green indicates phones that are neutral, neither associated with churn or not churn. Finally, brand names such as Blackberry were included to determine whether subscribers were complaining about a specific phone or a brand in general. (The orientation of the words is random.)

Figure 1: Word Cloud showing available phone models. Size of the words is correlated with the number of mentions (larger is more frequent). The color of the word indicates propensity to churn: red indicates more churn, blue indicates less churn and green is neutral. Orientation is random.

Using this strategy, the carrier was able to visualize and communicate how specific phone models were highly correlated with churn outcomes, while brand name mentions were neutral. Specifically, certain older Samsung and Blackberry phones were particularly correlated with churn outcomes. Mentions of the iPhone, though not offered by the carrier at the time of analysis, were also highly correlated with churn confirming their suspicion that subscribers desiring the iPhone were likely to leave the carrier. Understanding problematic phone models allowed the carrier to target subscribers using those phones with offers of free upgrades in order to retain their contracts.

Textual Models Improve Overall Accuracy

Prior to the introduction of analytic modeling to predict churn, the carrier used human intuition to identify likely churners. In a direct comparison, the structured data models proved to be 2.5 times more effective. By itself, the textual model was also about that effective. Using both

models together increased the overall effectiveness another 3.1 percent, a modest but statistically significant improvement.

The Rest of the Story

Shortly after this model was put into production, the iPhone was approved for use on the carrier's network. This led to an immediate growth in subscribers. Flush with new subscribers, the carrier reduced its focus on churn and the model was shelved. After a few months, however, the iPhone halo wore off and churn was actually happening at a higher rate than before the iPhone was introduced. Despite the temporary boost in new contracts, the network was still experiencing a net loss of customers.

The churn prediction model was quickly reinstated and contributed to a period of dramatic growth in the number of subscribers creating millions of dollars in additional revenue due to retained customers. As shown in Figure 2, this return has paid for the initial investment many times over (in fact, recouping the initial investment every month) and the clear positive results have sparked a larger emphasis on analytics across the company.

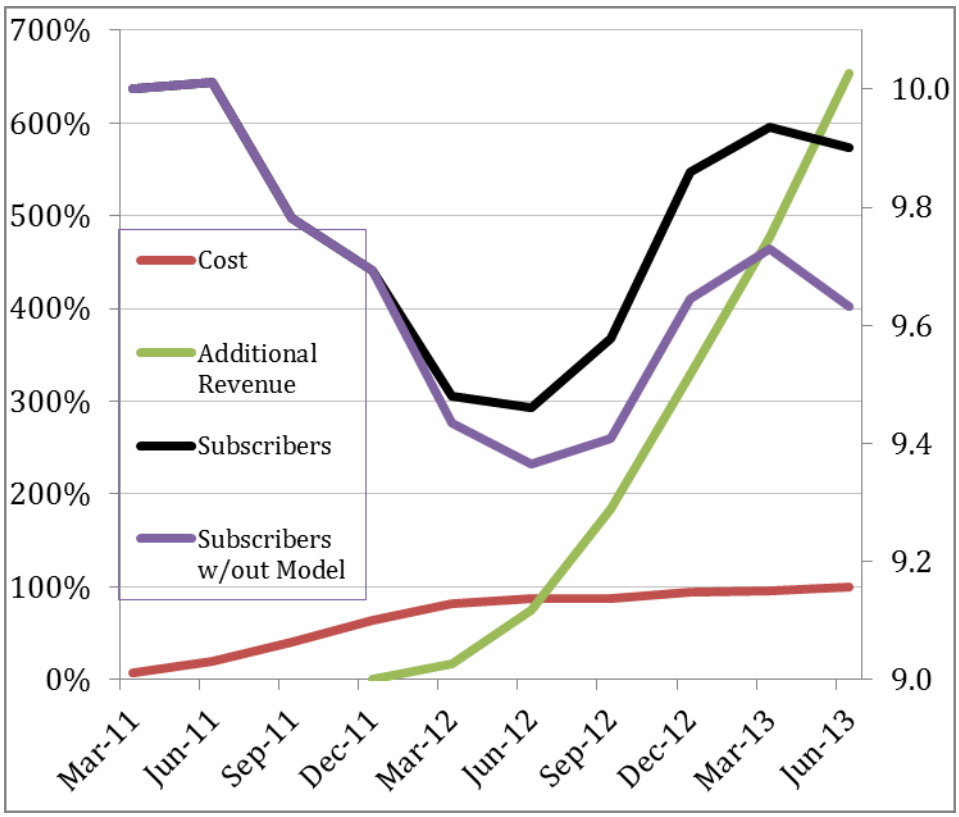

Figure 2: Results of the churn efforts measured in both cost (as percentage of analytics effort) and number of subscribers compared to a parallel campaign.

Technology for Text

Automated processing of text requires all the analytic skills required for structured data and a linguistic understanding of language to be successful. The scale and variability of textual data has resulted in a proliferation of software tools designed for text, each including different mixtures of numerical and linguistic processing. Part of the methodological diversity arises from there being two competing schools of thought - commonly referred to as Text Analytics and Text Mining. Below, we discuss these two areas briefly, and also recommend the type of software to use to solve problems similar to the mobile phone churn example described above.

Text Analytics vs. Text Mining

In his book *Thinking Fast and Slow*, Nobel laureate Daniel Kahneman describes two modes of thinking: a fast, reactive system Kahneman calls *System 1* and a more deliberative system Kahneman calls *System 2*. These two modes of thinking mirror two different approaches to the analytical processing of text. *Text Mining* is a set of techniques derived from statistics, data mining, and machine learning for automatically processing text using statistical methods. In contrast, *Text Analytics* automates a set of ideas and approaches from linguistics and emphasizes encoding human knowledge in an automated, repeatable format. Though there are passionate proponents of each, in head-to-head comparisons we have found both the statistical Text Mining approach and linguistic Text Analytics approach perform equally well.

Although performance is equivalent, the two approaches get to their results in different ways, which leads to different strengths and weaknesses. As a machine-driven approach, Text Mining mirrors Kahneman's System 1, producing results quickly, but can be overzealous and will occasionally produce nonsensical results. In contrast, Text Analytics generally produces high quality results, but may not generalize well to new data as human-generated rules tend to be more fragile than algorithmic rules. Like Kahneman's two systems, Text Mining and Text Analytics are complimentary and when combined can produce results that are better than either approach on its own. We will discuss strategies for combining these approaches in greater detail in a following research brief.

Software for Text

There are many available software packages for processing text — so many that choosing the right package can be a real challenge. Some tools are geared for search and knowledge

management applications, others for document classification and auto-tagging, and still others for information extraction. Each of these represents a different application area for text and is helpful for understanding the content of the text. We have found, however, that the maximum value can be extracted when the text is combined with structured data in a single predictive model.[1]

For predictive models, we recommend packages that combine textual processing and structured data processing. This facilitates combining text and structured data into a single model as described in the case study above. Each of the software packages rated as "leaders" in Gartner's Magic Quadrant in Analytics can combine text and structured data together. These include *SAS Enterprise Miner*, *IBM Modeler Premium*, *KNIME* and *RapidMiner*. Popular open-source tools such as *R* also provide mechanisms for effectively combining the two types of data into a single approach.

Despite the integration, none of the packages is a one-stop shop. There are simply too many different algorithms and methods applicable to text to include in a single software package.

Strategies for Successful Text Analytics Projects

Text analytics differs significantly from structured data analytics. The following are ways to achieve success with text despite its differences from typical predictive analytics.

1. **Keep expectations high, but be realistic and allow more time to achieve success**. Modeling unstructured text is harder than modeling structured data and has different requirements and approaches. Consequently, different tools and levels of effort are required to make text analytics be successful. Processing text especially requires more time for data preparation and processing.

2. **Text analytics requires a different set of skills than predictive analytics**. An example comes from a content provider who focused on deep knowledge management. To achieve their goals, the firm employed six taxonomists. Of the six taxonomists, all were women; all held advanced library science degrees, all enjoyed puzzles and brain teasers, and all but one were left-handed. We're not saying you need to find left-handed women with library science degrees to be successful with text analytics, but special care does need to be taken to find the right people for the job.

[1] Please see Chapter 2 of *Practical Text Mining and Statistical Analysis for Non-Structured Text Data Applications,* G. Miner, D. Delen, J. Elder, A. Fast, T. Hill, and R. Nisbet, Elsevier, January 2012

3. **Combining unstructured textual data and structured data into a single model is critical for success**. Text contains higher value information, but that information is context dependent and is difficult to extract effectively due to that ambiguity. Structured data is not ambiguous and can be used to draw out the full value of text. There are three methods for incorporating textual features into a structured model:
    a) *"Bag of Words"*: Here, words appearing in the document are added directly as input features to the predictive model. Techniques such as stopwords and stemming are used to ensure that common words are not over-represented.
    b) *Feature Extraction*: Here, specific words and phrases are extracted from documents to be included as features. Some type of natural language processing and/or concept extraction technique is applied to the individual words prior to feature creation.
    c) *Single Model Results*: With this approach, the text alone is used either to create a specific classification or to produce a numerical score (high-order text data feature) that is then used as an input into the structured data analytics. This has the advantage of allowing text processing separately from the numerical processing which opens up different tool possibilities, but it also precludes interactions between the textual and structured data features.

4. **Faceted search is an excellent way to make quick progress and combine textual and structured data.** Search provides an alternative method for browsing textual data. Search enables more extensive exploration as reading a sample of documents is usually not sufficient to get a complete view of a corpus. Search allows the data exploration and understanding necessary to develop advanced models of text. *Document facets* are structured metadata associated with a document that can be used to filter and refine keyword searches. For example, consider how an online shoe retailer might provide facets for size, color, or designer. These facets enhance the full text description of the shoes.

5. **Processing text at multiple levels is a best practice for building a textual model.** First, word level (and then phrase-level) processing can identify key concepts. Second, combine multiple related concepts (such as synonyms or abbreviations) into a single input feature to increase the understandability of the model. Third, use document-level processing to correlate the individual features with known business outcomes and produce a final score.

Conclusion

Unstructured textual data has the potential to provide significant value, which can be further enhanced by combining textual and structured data together in a single model. Text must be processed at both the word and document levels. For best results, solutions requiring textual inputs should use both human understanding and machine processing together.

Text Analytics and Text Mining are two different, but complementary, approaches for processing textual data, relying primarily on language-based rules vs. statistical discoveries, respectively. Passionate supporters of each approach argue for its superiority yet, the best solutions integrate strategies from both. We will explore the claims of Text Analytics and Text Mining proponents, and describe a successful approach that harnesses the strengths of both.

About the Authors

Andrew Fast, Ph.D. is the Chief Scientist at Elder Research and leads the research and development of new tools and algorithms for data and text mining. Dr. Fast graduated Magna Cum Laude from Bethel University and earned Master's and Ph.D. degrees in Computer Science from the University of Massachusetts Amherst. There, his research focused on causal data mining and mining complex relational data such as social networks. Dr. Fast has published on an array of applications including detecting securities fraud using the social network among brokers, and understanding the structure of criminal and violent groups. Other publications cover modeling peer-to-peer music file sharing networks, understanding how collective classification works, and predicting playoff success of NFL head coaches (work featured on ESPN.com). With colleague Dr. John Elder and others, Andrew has written a book on *Practical Text Mining* that was awarded the *PROSE* Award for Computer Science in 2012.

John Elder, Ph.D. has authored innovative data mining tools, is a frequent keynote speaker, and was co-chair of the 2009 Knowledge Discovery and Data Mining conference, in Paris. Dr. Elder co-authored 3 books (on practical data mining, ensembles, and text mining), two of which won "book of the year" awards in Mathematics or Computer Science. John's courses on analysis techniques -- taught at dozens of universities, companies, and government labs -- are noted for their clarity and effectiveness. Dr. Elder earned Engineering degrees from Rice and UVA and is an Adjunct Professor of Systems Engineering at UVA. He was honored to be named by President Bush to serve 5 years on a panel to guide technology for national security. Lastly, John is grateful to be a follower of Christ and the father of five.

Elder Research Inc. (ERI) is the US's leading consulting company in data mining, predictive analytics, and text mining. Founded by John Elder, in 1995, ERI has helped government agencies and Fortune Global 500® companies solve real-world problems by amplifying the productivity of their analysts. Drawing from experience in multiple industries, ERI brings cutting-edge technology into front-line practice to achieve high return on investment. Headquartered in Charlottesville, Virginia, ERI also has a growing office in Washington, DC.