SOCIETY FOR SCIENCE AND EDUCATION
UNITED KINGDOM

TMLAI TRANSACTIONS ON MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE

Volume 5 No. 3
ISSN 2054-7390

Isolating Natural Problem Environments in Unconstrained Natural Language Processing: Corruption and Skew

Charles Wong
Executive Intelligence, LLC
ccwong@bu.edu

ABSTRACT

This work examines the full range of commonly available natural language processors' behaviors in a natural, unconstrained, and unguided environment. While permissible for typical research to constrain the language environment and to use in-depth knowledge to guide the processor for enhanced accuracy, this work purposefully avoids a clean laboratory in favor of a natural, chaotic, and uncontrollable environment. This shifts the focus towards natural processor behaviors in natural, unknown environments. This work provides a standardized comparison framework to compare and contrast each of a full range of processors' theoretical strengths. It continues to examine empirical behaviors on a full range of environments from typically used baseline sample documents, to actual raw natural texts used in an intent marketing business, to a series of increasingly corrupted and inconsistent sample documents to further differentiate processor behaviors. In all cases, the texts are unconstrained and the processors operate in their most naïve, default forms. Results complement and extend prior work. It adds that accuracy-centric processors like artificial neural networks or support vector machines require both highly constrained environments and in-depth knowledge of the processor to operate. Descriptive-centric processors like k-nearest neighbors, Rocchio, and naïve Bayes require only highly constrained environments. An explanatory-centric neurocognitive processor like Adaptive Resonance Theory can operate robustly with neither environmental constraint nor in-depth processing knowledge, but exposes operations to basic human temporal neurocognitive behaviors.

Keywords. Natural language processing; skew; corruption; natural behavior; neural networks; intent

ACKNOWLEDGEMENTS

This work represents the sole original effort of the author. No other individual, organization, or fund was involved in the making of this work.

1 Introduction

Processing refers to the reduction and constraining of a target activity into logical, controlled, and systematic machinery, typically for mass production. Natural language refers to the everyday vernacular of unconstrained human speech and text. The attached “natural” adjective serves only to differentiate it from the newer logical machine concept of language. Putting it together, natural language processing

DOI: 10.14738/tmlai.53.3229
Publication Date: 24th June, 2017
URL: http://dx.doi.org/10.14738/tmlai.53.3229

seeks to apply machine logical constraints to mass produce the reading and labeling of unconstrained language. Putting it together results in a juxtaposition of incompatible environments. Logical machine processing requires a closed logical environment for systematization. Natural language requires an open natural environment for expression.

This study bridges the gap by first categorizing machine learning processing efforts into discrete high-level processor families, each one of which emphasizes a particular goal. The second objective is to isolate and analyze a series of natural language characteristics that appear in natural environments but are particularly challenging to machine processing.

The processors in this study include the artificial neural network, the support vector machine, k-nearest neighbors, Rocchio, naïve Bayes, and an Adaptive Resonance Theory processor. The natural language environments include samples with poor and missing documents, skewed document distributions, falsely skewed document distributions, mislabeled documents, and dual-labeled duplicate documents. Poor document sample quality is a function not only of quantitative insufficiency but of document non-representativeness. A low ratio of documents to labels, for example, is one indicator. Other indicators include but are not limited to: mismatched vocabularies across the sample or high and changing rates of inconsistencies resulting in skews and corruption.

In the following section 2, this study reviews the prior literature and highlights the progression towards more complexity in natural language processors. In section 3, this study explores the background etymological approach for each processor family. Section 3 also provides details in exploring natural language environments. Section 4 presents results and discussion. Section 5 provides concluding remarks.

2 Review of natural language processing work

This review first needs standardized definitions. Since the typical first goal of machine processing of natural language text is to classify it towards one or more predefined labels, the literature often terms processor and processing as classifier and classification. The more machine learning and computational oriented research would describe the process thusly: translate the words of a text document into the numerically coded features or dimensions of a vector. Script, set up, or otherwise train the machine learning classifier on a sample of vectors, each of which is pre-assigned by a human expert to a class. Finally, classify additional, unknown vectors by their best fit to the classifier. Statistical or linguistic oriented research uses different terminology, though the procedure is similar.

For consistency, this study maintains a business-oriented natural language document terminology whereby sample documents containing words are pre-labeled by human experts. The processors evaulate the word vocabularies and set up based on this expert effort, then attempt to label additional unknown documents at scale in the same manner the experts had labeled the sample documents.

Nigam, et al., (1998) [1] used Expectation Maximization with a naïve Bayes processor in a self-supervised learning scheme to enhance accuracy on large groups of documents. The naïve Bayes processor first processed a small sample of expert-labeled documents. The processor then processed an additional batch of unlabeled documents using the Expectation Maximization algorithm to further improve accuracy. Documents included the full Newsgroup 20 database, using 20,017 documents over 20 labels in a documents/labels ratio of over 1,000. A second set of documents also included the WebKB dataset with 4,199 documents over 4 labels in a ratio in excess of 1,000. A third set of documents also included the

Reuters-ModApte database with 12,902 documents over the top ten labels, resulting in a ratio again in excess of 1,000. Results showed that removing stop words – e.g. commonly used grammatical words like, "at" "the" "of" or "from" – surprisingly impaired performance. Additional results showed that using higher documents/labels ratios provided better processing, with end result accuracy ranging from 20% with ratios of 1, to 65% with 100 samples per label, and up to 80% at ratios exceeding 250. In any case, including 500 or more unlabeled documents in additional processing improved the end result, with more benefits accruing on the weak performers (20% accuracy improved to 35%) and virtually no improvement on strong performing processors (80% accuracy showed no improvement).

Roussinov and Chen [2] explored whether text document clustering processors can closely mimick how well a human clusters documents. The experimental setup included 80 electronic meeting comments sorted into 8-10 labels. The study used either Wards Clustering or a Kohonen Clustering processor to automatically form the label clusters. The results then present to 17 human volunteers to post-process while comparing the level of effort required to correct clustering errors. Results showed that Wards Clustering was more precise, though the authors conclude this may be due to an artifact of implementing Wards Clustering to require keywords in common during clustering.

Yang and Liu [3] compared the performance of support vector machines, *k*-nearest neighbors, artificial neural networks, naïve Bayes, and regression natural language processors. Data included the Reuters ModApte dataset using 7,769 labeled documents sorted into 90 labels. This resulted in documents/labels ratio of 119. Results showed that the support vector machine and *k*-nearest neighbor processors performed best at 86% accuracy, with artificial neural networks scoring second best at 83% and naïve Bayes scoring 80%.

Sebastiani [4] compared a variety of natural language processors, including naïve Bayes, regression as a classifier, Rocchio, artificial neural networks, support vector machines, and *k*-nearest neighbors. Data included five different publicly available Reuters document sets: Reuters-22173 ModLewis, Reuters-22173 ModApte, Reuters-22173 ModWiener, Reuters-21578 ModApte, and Reuters-21578 ModApte. The resulting documents/labels ratios ranged from a low of 100 to a high of 900. Results showed the support vector machine produced the highest accuracy at 87%, with *k*-nearest neighbors and regression at a close second at 86%, and naïve Bayes and Rocchio ranking last at 79%.

Jing, et al. [5] evaluated different word valuation schemes in combination with a naïve Bayes processor. The evaluation included 9,603 sample documents sorted into 135 labels, resulting in a documents/labels ratio of 71. Results showed that a term frequency/mutual information vectorization produced results up to 88% accuracy. In contrast, the more traditional term frequency/inverse document frequency vectorization produced up to 76% accuracy.

Mittermayer [6] applied a support vector machine on a 3-label sentiment processor on 6,602 press release documents. The three labels included Good, Bad, and Neutral sentiments. However, due to the extreme skew in the data, with nearly 90% of the documents showing Neutral sentiment, the support vector machine required the document set to be rebalanced with uniform 200 documents per label using under-sampling (600 documents total). The results showed that the support vector machine developed on the artificially uniform rebalanced sample set provided very high precision, but poor recall regarding the Good and Bad sentiment labeling during testing. Since the sample set was uniform in distribution, the

processing results during testing were also nearly uniformly distributed. Results were still significantly enhanced over random guessing.

Mooney and Roy [7] applied a Naïve Bayes natural language processor on 3371 labeled documents over 2 labels in a positive or negative sentiment analysis. The documents/labels ratio was in excess of 1000. Results showed the Naïve Bayes processor processor scored up to 85% accuracy.

Ikonomakis, et al. [8] provided a review of literature for natural language processors. Results showed that the most commonly used processor processor included naïve Bayes, support vector machines, artificial neural networks, *k*-nearest neighbors, and mixtures of experts. General findings included that naïve Bayes are simple processors that are effective yet do not process natural language very well. Support vector machines tended to have very high precision but relatively poor recall.

Kim, Howland, and Park [9] focused on word reduction as a means of improving natural language processing accuracy. They first applied centroids, orthogonal centroids, and finally Latent Dirichlet Allocation pre-processing to reduce the number of words in each labeled sample document to a manageable number of key words. The reduced documents then fed into one of several processors for comparison. The processors included *k*-nearest neighbors and support vector machines. The labeled documents included a MEDLINE database with 2,500 sample documents sorted into 5 different labels, resulting in a documents/labels ratio of 500. Documents also included the Reuters ModApte dataset using 7,769 sample documents sorted into 90 labels, resulting in a ratio of 119. Results showed that the support vector machine processor with word reduction produced the highest labeling accuracy at 89%. The next best processor scored 88% accuracy.

Lan, et al. [10] compared a variety of word valuation schemes in conjunction with a support vector machine. Since machine processors can only operate on numeric values, the processor translates all words into a set of values. These values could range from a simple Boolean denoting the presence of a certain word to a complex analog value denoting the frequency of a given word as a proportion of all words in a document. Labeled documents included the Reuters-21578 set filtered to use only the top ten most common labels to remove the skew in the document labeling and to maximize the documents/labels ratio. Data also included the Newsgroup 20 set, with 300 documents for each of 20 labels – hence, a ratio of 300. Results showed that the novel term frequency/relevance frequency valuation produced the best results at 64% averaged F1 score. The standard term frequency valuation produced second best results at 60% averaged F1. The simplified Boolean valuation produced the weakest results at 56% average F1. F1 scores refer to the precision and recall scores averaged across all labels.

Zhang, Yoshida, and Tang [11] explored compressing documents into extracted multi-word phrases for processing with support vector machines. The documents included approximately 509 sample documents across four labels derived from the Reuters-21578 document set. The selection centered on documents that could generate meaningful multi-word phrases. An additional 254 documents were reserved for evaluation. The documents/labels ratio was 127. Results showed that using individual words outperformed multi-word phrases, with 91% accuracy over 87%. Further, a support vector machine using a linear kernel outperforms a non-linear kernel.

Ko [12] explored a novel word valuation scheme, the term relevance ratio, in conjunction with a support vector machine and *k*-nearest neighbors. The evaluation used the Reuters-ModApte database with 12,902 documents over the top 10 labels and the Newsgroup 20 database with 20,017 documents over 20 labels,

both resulting in a documents/labels ratios over 1,000. The term relevance ratio word valuation scheme used probability estimations of a given word's distribution across labels rather than simply its frequency across all documents. Results showed that the novel word valuation scheme improved accuracy from 92-94% up to 95% across both processors.

Colace, et al. [13] use a probabilistic topic model, itself based on the Latent Dirichlet Allocation, to extract more informative word pairs from sample documents. The evaluation used the Reuters-ModApte database with 9,603 sample documents over the top 10 most numerous topics resulting in a documents/labels ratio of 960. Results showed a macro-averaged F1 score of 30% when using all training samples, but up to 75% F1 score when using only a small (1%) random fraction of the sample documents. Their processor became less discriminatory with greater numbers of sample documents.

Based on these studies, several broad conclusions stand out. First, the most common natural language processors are support vector machines, naïve Bayes, *k*-nearest neighbors, and artificial neural networks. Second, these processor performances are strongly dependent on complex and extensive document pre-processing and filtering to provide a controlled laboratory environment.

The processors required vast amounts of high quality, static labeled sample documents resulting in documents/labels ratios in excess of 100. The documents also tend to be manually and artificially de-skewed, rebalanced, and highly processed, resulting in highly uniform and homoskedastic samples - that is, where any sufficiently large partition of the sample fully represents the whole. This is necessary in the course of logically and controllably evaluating an isolated processor's potential capabilities.

In contrast, this study aims to explore natural language processors in a purposefully unconstrained and chaotic natural environment. The operators setting up the processors may lack the in-depth working knowledge that the scientist would acquire from designing it. The document samples may be limited, skewed, and even heteroskedastic - that is, where even sufficiently large partitions of the sample do not represent the whole. These conditions unfortunately often describe the natural language environment, with the intent marketing business as an apt example.

The following section details the background development and philosophies for the commonly used natural language. It also describes the characteristics of a natural language environment as might be encountered in intent marketing conditions.

3 Natural language processing families and the natural language environment

3.1 Natural language processing families.

Based on a review of the literature, the most commonly used natural language processors are: artificial neural networks, support vector machines, *k*-nearest neighbors, Rocchio, and naïve Bayes. These are commonly known as machine learning computational or statistical processors. To provide an additional human cognitive perspective, this study includes a neuro-cognitive processor based on Adaptive Resonance Theory. Since these processors span different mathematical, statistical, and neuropsychological fields, this study will describe each not in terms of their particular disparate etymological concepts but rather in their philosophical relation to each other in common, practical terms. The description focuses on what each processor physically stores to represent labeled documents.

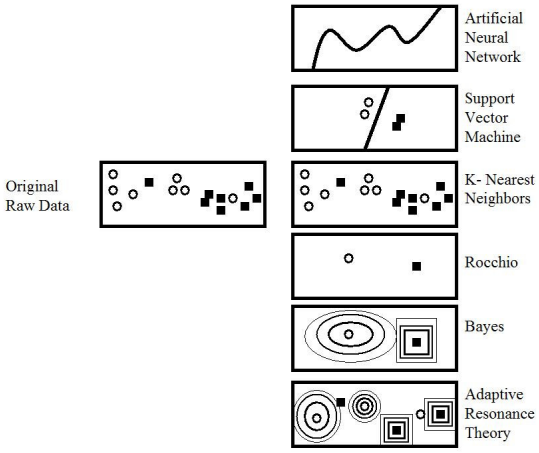

Figure 1

Six different processor family approaches towards processing and storing the same set of documents during setup. The original raw data documents are a simplified 2-label, 2-dimensional set for demonstration (left). Eight white circles represent eight similar documents. Eight black squares represent another set of eight documents with a different label. An artificial neural network stores a flexible, weighted dividing boundary that best summarizes all 16 documents. A support vector machine stores only the optimal key documents and forms the optimal label boundary. Rocchio stores two abstract, manufactured summary documents, one for each label. A k-nearest neighbor stores all 16 documents as is. Naïve Bayes stores two abstract, manufactured summary documents, one per label, plus the weighted probability that any given word draws towards a particular label. The document boundaries are standard deviations dependent on word and processor setup; the different circular or square shapes are only represented here for visual contrast. Adaptive Resonance Theory stores abstract, manufactured summary documents and the weighted probability that any given word draws towards a particular label. However, Adaptive Resonance Theory decides endogenously how to further partition sub-labels with additional, locally manufactured summary documents.

Artificial neural networks (ANN) [14] in their incarnation as a family of mathematical functions are an extension of the Perceptron [15]. A single Perceptron compresses all documents with a given label into a unified meta-document. It then finds a linear dividing boundary between these meta-documents. Were a Perceptron in Figure 1, it would draw a single diagonal line. ANNs combine a lattice of Perceptrons in parallel and series units, resulting in a potentially highly curved and warped dividing boundary, as shown in Figure 1. ANNs are also known as multi-layered Perceptrons. Individual variance among its internal Perceptron units is typically stochastic. That is, each parallel Perceptron unit begins with randomly assigned biases to help it specialize on particular words that may help differentiate meta-documents. The units later in the series can re-process to combine multiple meta-documents. For example, given 16 documents evenly sorted into 2 labels, an ANN with 10 units behaves as 10 stochastically differentiated specialists with confidence-weighted voting each operating on different aspects of 2 meta-documents. Each unit might randomly specialize to detect all words in a particular label. Or they might randomly specialize on a particular set of key words regardless of label. More likely, each blends both approaches, typically with large overlaps with neighboring units. The exact purpose for each unit cannot be determined in advance. The ANN processor has the complete freedom to assign each unit's purpose.

ANNs are loosely mathematically and biologically inspired to focus on accuracy. While there are many different forms of ANNs, the primary factor is the number and structure of the units. Processor setup requires the operator to have deep a priori knowledge of the natural language sample and environment to determine the optimal number of units. Alternatively, the operator may run and re-run the ANN with varying numbers of units. Due to the stochastic nature, multiple re-runs are necessary. The default form here uses an arbitrary 10 units in a single parallel layer.

Support vector machines (SVM) [16] as a family of mathematically optimized processors are also an extension of the Perceptron. While ANNs emphasize an additive expanded ensemble of multiple Perceptron units with a curved boundary for accuracy, SVMs emphasize a subtractive document reduction (i.e. support vector reduction) to optimally place the linear boundary. By effectively reducing and eliminating non-informative non-supporting documents from consideration, an SVM can optimally fit the linear boundary onto a manageable set of optimal key sample documents. For example, given 16 documents evenly sorted into 2 labels, an SVM might only consider the 4 most poignant key documents and ignore the remainder. The SVM generates the optimally angled linear boundary based on these key documents. The document reduction and the linear boundary placement are mutually dependent. This results in the processor being optimally set up in theory.

SVMs are mathematically inspired to focus on accuracy. While there are many variations on the SVM, the primary factor is the kernel, or transfer function. The kernel dictates how the reduction and optimal boundary interact. This kernel is fixed. The operator must pre-design this selection a priori. The operator needs deep knowledge of the natural language environment to select appropriately. Adjusting the kernel has a dramatic impact on processor behavior. The default form here uses a non-linear Gaussian kernel function per the publicly available LibSVM package (http://www.csie.ntu.edu.tw/~cjlin/libsvm/).

k-nearest neighbors [17] is a statistical processor that dispenses with forming the boundary. It can be viewed as the polar opposite to an SVM as it has neither any discrete boundary nor any reduction in documents. It simply stores all documents as is. For example, given 16 documents evenly sorted into 2 labels, a KNN stores all 16 documents. It processes unknown documents by matching them against *k* out of these 16 documents and taking a majority vote.

KNNs emphasize describing the samples with no interpretation. While there are many variants on the KNN, the primary setting is *k*, the size of the subset of stored documents to consider for voting. This *k* parameter is operator selected in advance by the operator and can significantly affect the natural language processing based on the underlying patterns in the natural language environment. The default form here uses *k*=1.

Rocchio is a statistical and visualization processor with no discrete boundaries. Rocchio compresses all similarly labeled documents into a single summary meta-document. It does not form any boundary. For example, given 16 documents evenly sorted into 2 labels, Rocchio produces 2 meta-documents. It processes unknown documents by matching it to the nearest meta-document.

Rocchio focuses on a descriptive summary of the samples. Rocchio stores the documents in an extremely easy-to-visualize manner. While there are varieties of Rocchio-like processors, the primary identifying

factor is how it compresses meta-documents. In this study, Rocchio uses a simple, fixed centroid showing the average of word occurences. There are no additional adjustable settings by the operator.

Naïve Bayes [17] is a statistical processor with probabilistic boundaries. It compresses all similarly labeled documents into a single summary meta-document. Around each meta-document, naïve Bayes also stores fuzzy, probabilistic boundaries based on the standard deviations of the word frequencies. For example, given 16 documents evenly sorted into 2 labels, a naïve Bayes stores 2 meta-documents with standard deviations across all words. Rather than matching unknown documents to the nearest meta-document based strictly on number of words in common, a Naïve Bayes takes into account the probabilistic distribution of these words to fine tune its probabilistic match.

Naïve Bayes balances descriptiveness and accuracy. There are a variety of Bayesian processors, dependent on how they form meta-documents. In this study, it forms single average meta-documents with standard deviations. There are minimal parameters for modification by operator. By including probabilistic boundaries, naïve Bayes is able to extract more word distribution information but not at depths sufficient to warrant extensive manual operator assistance.

Since this study examines the juxtaposition of processors with natural language, it includes one additional processor family. Adaptive Resonance Theory (ART) is a neural-cognitive approach that readily models natural behavior [18,19] and can therefore apply to natural language processing. ART focuses on hippocampal memory formation. Recent work shows that even adult brains exhibit hippocampal new nerve generation [20], especially given novel environments [21,22]. The core of the ART processor is adding or assigning new memory cells upon detection of anomalies and novelty. For example, given 16 documents evenly sorted into 2 labels, ART will attempt to store 2 summary meta-documents with word-frequency boundaries. Should it encounter documents that unexpectedly blur or confound these boundaries, it triggers the anomaly detection and creates additional sub-label mini-meta-documents.

ART is focused on neither accuracy nor descriptiveness. It does not attempt to borrow biological concepts in a bid to maximize accuracy. Rather, it attempts to explain the cognitive and biological concepts themselves. There are no free parameters since this would require an exogenous operator. There is no stochastic randomness since this would provide no explanatory power.

The next section explores the characteristics of text documents in a natural language setting as opposed to a cleaned, static lab setting.

3.2 Natural language environment.

In the intent marketing business, the goal is to attract potential sales by engaging the consumer. Given the consumer's currently selected content - e.g. article, advertisement, or other document - the business recommends another related document likely to satisfy the consumer's implied intent or goal. Doing so builds trust and rapport with the consumer, thereby lowering the threshold for future sales transactions. Therefore, the basic underlying task is to be able to match related documents together.

The complexity arises from the nature of natural language. What are the formats for the content? By definition, natural language content has no format. What are the possible intent labels? Can they overlap? Can they change? What is the content sample? Can more be added? What is the context? These can generate implacable complications. For example, the commonly used term frequency/inverse document frequency word valuation that translates text to numeric values uses the ratio of word occurrences in the

document over the total numbers of documents in which it occurs. This value is straightforward in a constrained, logical sample with a cap on the number of documents. It is undefined with no denominator in an uncontrolled, changing natural environment where the total number of documents can be variable.

Given these complications, businesses in natural environments use extensive human expert staff to manually label sample documents as they expect consumers would. This generates consumer-level high quality intent-matching recommendations. However, to continually track changing consumer intents with multiple experts, the labeling effort necessarily may be skewed, inconsistent, overlapping, and changing. To otherwise constrain the labeling effort at this point is to fundamentally break with the consumer. A follow on machine processor attempts to systematize this expert human effort, but must deal with skews, inconsistencies, corruptions, and changing environments to do so.

This study starts with the commonly used Reuters-21578 document set with the r8 subset division. It includes 5,485 documents over 8 labels with a documents/labels ratio of 685. Boolean tags represent the words in each document. That is, the word valuation can be either 1 or 0 for each word in each document. According to the literature, this Boolean valuation is the very simplest and worst performing scheme. This study finds this scheme suitable since it involves minimal environmental transformation or constraining into a more logical and unnatural form. It maintains the document environment in its most challenging, simplest natural raw form.

Using the Reuters-21578 r8 document set, this study generates the following case scenarios using subsets to explicitly replicate raw, imperfect, skewed, and corrupted natural conditions:

(1) Baseline case full. Uses all 5,485 sample documents representing the full sample to best replicate earlier research, but with minimal environmental cleansing or processor tuning. The reserved test set contains 2,189 additional documents.
(2) Baseline case partial. Uses 1,000 sample documents representing limited sample data available. The reserved test set contains the 2,189 additional documents.
(3) Minor human error. A 5% portion of the sample documents from (2) is randomly mislabeled to “Error” to represent human expert error. The reserved test set contains the 2,189 additional documents.
(4) Major human error. A 15% portion of the sample documents from (2) is randomly mislabeled to “Error” to represent human expert error. The reserved test set contains the 2,189 additional documents.
(5) Logical clean environment. This case contains 20 documents for each label, totaling 160 documents. The reserved test set is the same exact sample. This represents a hypothetical scenario where the data is guaranteed to be uniform, stationary, and representative. Testing on the identical training set guarantees it is representative.
(6) False skewed sample. Using the De-skewed set (5), this set duplicates all 20 documents from one random label by a factor of 5. This results in 240 sample documents for setup, with 7 labels containing 20 documents and 1 label containing 100. The reserved test set is the same as from (5).
(7) Early data corruption, Minor. Using the De-skewed set (5), this set duplicates 5 documents (25%) from each label. This set of duplicates has their labels swapped to represent the worst form of data corruption – identical documents with different labels. These erroneous duplicates are randomly shuffled and inserted near the beginning of the data set. This results in 200 sample documents, with 25 documents assigned to each label. The test set is the same as from (5).
(8) Early data corruption, major. Using the De-skewed set (5), this set duplicates 10 documents (50%) from each label. This set of duplicates has their class labels swapped to represent the worst form of data corruption – duplicate documents with different labels. These erroneous duplicates are randomly shuffled and inserted near the beginning of the data set. This results in 240 sample documents, with 30 documents assigned to each label. The test set is the same as from (5).
(9) Late data corruption, Minor. Follows the same setup as (7), but with the erroneous duplicated documents inserted near the end of the setup sample.

(10) Late data corruption, major. Follows the same setup as (8), but with the erroneous duplicated documents inserted near the end of the setup sample.

Scenarios (2)-(10) reflect the more realistic environment where the average ratio of documents/labels in a natural language intent marketing business is far less than what is typically packaged and made publicly available for research studies. Here, the ratios range from 100 down to 5. Since a common business requirement is that the number of labels may grow, it is not uncommon to experience low ratios.

Scenarios (7)-(10) are especially pernicious since these represent theoretically the worst possible scenario for a natural language processor. The mislabeled duplicates cannot be separated. Since they have different labels, these are high-information value designators for the boundaries between labels and cannot be ignored. These scenarios can occur when multiple experts work together to assign labels or when two different document sources are combined, a fairly common occurrence in business. Scenarios (7) and (8) represent the more common scenario with earlier labeling work being less reliable. Since businesses are constantly evolving with changing goals and employee skill sets, it is fairly common for work quality and business matching to become progressively higher quality and more complete over time. Earlier work may not accurately represent later, more current needs and environments. Scenarios (9) and (10) represent a less common business scenario where the organization has suffered in quality over time and has become less reliable. It presents here to better examine the temporal picture.

In addition, one final case scenario (11) includes an aggregated document set combined from an intent marketing business to represent the natural environment. This contains 4,000 sample setup documents over 1,440 labels resulting in a documents/labels ratio of less than 4. The sampled documents originated from 56 different human experts evenly spanning a non-stationary 4-year business period. As the business expanded and changed, more experts were added, more intent labels were added, and the intent labeling emphasis shifted to capture revenue-generation trends. This shift caused heavy skews, with 310 documents sharing the most common label and 263 documents containing unique labels. Examination of the data also shows duplications and extensive (>15%) mislabeling, especially concentrated on earlier date-stamped work.

In all cases, the analysis here includes both raw test document accuracy and the macro-averaged F1 scores that harmonize the averaged precision and recall values for common comparison with a variety of prior work.

4 Results and Discussion.

Again, the purpose of this study is not to determine the best overall or particular winner. Each natural language processor family demonstrates a particular strength. The purpose of this study is to establish a framework to explore these strengths in a diverse natural language environment.

Table 1 shows the results of six different natural language processor families over 11 different environments. The first score shows the macro-averaged F1 scores. The second score in parentheses shows the pure accuracy score for comparison and baseline vis-à-vis reports in the prior literature. In each processor column, bold-faced and boxed values show under which scenario the processor best reveals its strengths. The shaded value highlights where that processor had its most relative difficulty. These strengths and difficulties are relative to its own column performance across scenarios and row considering its uniqueness to other processors.

Table 1. Six natural language processor family strengths and weaknesses over 11 different natural language environments.

Case	Notes	Sample	Testing	Labels	ANN	SVM	KNN	Rocchio	Bayes	ART
1	Baseline, full	5485	2189	8	26.1% (51%)	8.3% (49%)	41.5% (84%)	18.1% (60%)	24.5% (82%)	48.0% (84%)
2	Baseline, partial	1000	2189	8	14.2% (27%)	8.3% (49%)	36.5% (78%)	15.3% (45%)	17.8% (76%)	44.0% (81%)
3	Minor Human error	1000	2189	8	14.7% (26%)	8.3% (49%)	33.5% (74%)	18.2% (33%)	13.1% (68%)	45.5% (74%)
4	Major Human error	1000	2189	8	11.2% (21%)	8.3% (49%)	31.7% (69%)	18.2% (31%)	13.3% (65%)	41.6% (72%)
5	Logical Clean	160	160	8	**83.2% (90%)**	86.7% (89%)	100% (100%)	37.1% (76%)	96.8% (97%)	100% (100%)
6	False Skewed	240	160	8	2.7% (13%)	2.7% (13%)	**100% (100%)**	**37.1% (76%)**	8.6% (16%)	**100% (100%)**
7	Early Corruption, Minor	200	160	8	16.9% (23%)	**84.7% (86%)**	75.0% (75%)	14.3% (30%)	**92.5% (93%)**	**98.1% (98%)**
8	Early Corruption, Major	240	160	8	2.7% (8%)	22.5% (25%)	49.4% (49%)	9.4% (23%)	77.9% (79%)	**93.7% (94%)**
9	Late Corruption, Minor	200	160	8	16.9% (23%)	84.7% (86%)	75.0% (75%)	14.3% (30%)	92.5% (93%)	75.6% (76%)
10	Late Corruption, Major	240	160	8	2.7% (8%)	22.5% (25%)	49.4% (49%)	9.4% (23%)	77.9% (79%)	50.0% (50%)
11	In vivo Business Data	4000	1381	1440	3.1% (9%)	3.7% (9%)	17.3% (29%)	6.4% (17%)	13.3% (37%)	38.6% (57%)

Table 2 summarizes the six families based on their specific goal, and their specific level of concern regarding their natural language environment and their processor operations. Their environment concern refers to how much an impact that document skew or corruption for example can have on the processor. Their operations concern refers to the processor set up and maintenance. These are a function of the number of variables, and the time and skill level required to successfully adjust them. Concern or risks rise if minor adjustments in any single variable results in major behavioral changes. The results of Table 2 complement and extend the results of Table 1.

Table 2. Contrasting six different natural language processor families' emphases, environmental formatting, and operational requirements.

	ANN	SVM	KNN	Rocchio	Bayes	ART
Emphasis	Accurate	Accurate	Descriptive	Descriptive	Descriptive	NeuroCognitive
Environment Concern	High	High	High	High	High	Minimal
Environment Problem	Corruption	Skew	Corruption	Corruption	Skew	Cognitive Bias
Operations Concern	High	High	Low	Minimal	Minimal	Minimal
Operations Variable	Units	Kernel	Search breadth	N/A	N/A	N/A

In this study, the ANN and SVM processors focus on accuracy. They accomplish their stated goals without regard for any other consideration. This design philosophy can freely add additional complexity and commensurate required skill level and resources to operate so long as it can demonstrate its accuracy. This accounts for their "blackbox" like behavior where highly tuned and accurate ANNs and SVMs often cannot explain precisely why they behave as they do. This is also borne out by the sheer number of highly tunable and high impact operational variables. Adjusting any one of them slightly can have dramatic differences in ultimate behavior. Their most sensitive variables are the number of units and the kernel, respectively.

This partially accounts for the discrepancy between prior published accuracy results for SVMs and ANNs, typically among the best performing in the literature. In this study, the ANN and SVM were operated in arbitrary off-the-shelf default mode in as much as possible, reflecting a most naïve approach from an unskilled operator. For example, prior work [11] showed that an SVM tuned with a linear kernel produces better results than with a non-linear kernel on natural language documents. However, the publicly available LibSVM package also used here defaults with a non-linear Gaussian kernel. All processors in this study operate without any foreknowledge that they will operate on a natural language environment. The

remaining discrepancy may arise from the minimal cleansing and lack of constraints on the natural language environment.

Both the ANN and the SVM performed strongly under test case (5), in which the documents are de-skewed, cleansed, and tested on the same documents used during setup. This replicates the best possible condition with unskilled operators. It necessarily guarantees all documents are labeled correctly, are perfectly balanced, and precisely representative of the controlled test environment. This shows the untuned processors can function appropriately under a static, cleansed lab environment. This contrasts with test cases (6-10), with purposely falsely skewed labels and increasing levels of document corruption. The ANN showed relatively more vulnerability to corruption while the SVM showed more vulnerability to the skew. The natural business data test cases (1-4) and (11) with known corruption and skews were also significant challenges to the unmodified ANN and SVM.

KNN, Rocchio, and naïve Bayes share their emphases on descriptiveness. Rather than accomplish a processing accuracy measure by any means possible, these families focus on describing and summarizing the documents. Description-centric processors can more simply address a variety of business end goals and are more robust to poor operator skill sets. Only kNN has any meaningful adjustable variables, but results appear strong in the default, off-the-shelf intuitive setting with k=1.

In this study replicating unskilled, naïve operators on natural, unconstrained environments, the KNN, Rocchio, and naïve Bayes generated accuracies similar to the published figures per test case (1). While all can perform well on the artificially controlled test case (5), KNN and Rocchio can demonstrate robust performance on test case (6) – the false skew condition – and relatively weak performance on test case (8) – the major corruption condition. This shows that KNN and Rocchio are relatively resistant to skew, but vulnerable to mislabeled documents. This is due in part because both KNN and Rocchio store their documents independently of each other, thereby mitigating skewed distributions' effects from one label to another. This independence in turn leaves it exposed to corruption since it cannot consider other example documents to smooth out perturbations. Naïve Bayes shows the opposite strength characteristics, with strong resistance to minor levels of document labeling corruption, but vulnerability to skew. Its use of probabilistic deviation information smoothes out corrupt perturbations, but forces it to consider skewed distributions in generating its results.

ART emphasizes neurocognitive explanation. It attempts to explain human memory and behavior in a natural environment with its imperfections. In short, rather than translating, cleansing, and constraining the natural language into logical processor terms, it strives to remain more fully in the natural environment without regard for perfection.

ART best showed its strengths under the false skew (6) and under the early major corruption (8) scenarios. It showed the most difficulty under the late major corruption scenario (10). This can be explained by ART operations as shown in figure 1. It retains deviation information about each meta-document, leading to corruption resistance. It also reserves the ability to independently split meta-documents to segment and encapsulate skewed distributions. This allows it to de-emphasize and isolate anomalous documents while still retaining its information. It makes fewer assumptions about the document environment.

ART does have a unique weakness however, in that the timing of the corruptions has an impact that no other processor in this study shows. While extremely robust to corruptions encountered early, more recent corruptions of an identical form pose a relatively greater adverse impact.

To explain, this study highlights the strongest possible form of document labeling corruption – that of identical documents possessing different labels. Aside from expert labeling error, this occurs in natural environments as conditions naturally change. As an intuitive example, the same document about margarine could be correctly labeled "healthy diet food" or alternately "unhealthy junk food" depending on its labeling date. Modern consumers today now know that the trans-fat content of margarine is "unhealthy junk food," but this was not always the correct case. Margarine was originally designed to reduce saturated fat so as to be a healthy diet food. It was only discovered later that the trans-fat replacement had an even worse health effect.

Per Figure 1, if an ART processor first observes a document on margarine getting a "healthy diet food" label, it groups that margarine document with other "healthy diet food" documents into a meta-document. When it later observes an identical margarine document getting a different, "unhealthy junk food" label, it responds by creating a new sub-level mini-meta document. The new sub-level mini-meta document is placed with boundaries such that subsequent documents specifically about margarine fall under the "unhealthy junk food" label. This replicates how a modern day consumer would respond to margarine and this accounts for its uniquely robust performance in cases (7-8) where it reflects that the state of margarine has changed in the natural environment. Cases (9-10) represent the case where the state of margarine has not changed (i.e. margarine is still truly a "healthy diet food") and that the new directives from management (i.e. recent expert labels consistently showing "unhealthy junk food") are in fact false information. While these document scenarios (9-10) are atypical in natural and intent marketing environments, it presents here to demonstrate ART's primary weakness of temporality. If the order of sample documentation presentation erroneously trends away from the true labels, an ART natural language processor uniquely anticipates and extends the trend astray.

5 Concluding Remarks.

In contrast to prior research, this study asks not which logical processor is optimal in a transformed and constrained logical language environment. Rather, it asks how processors in their natural automated state behave in an unconstrained natural language environment. To do this, the study first needed to identify how and why the common processor families behave. The study also needed to identify natural language characteristics that provide particular challenges to automated processing. Results showed that accuracy-centric processors relying on mathematical and computational approaches require extensive document pre-processing and translation into a cleansed environment with extensive processor tuning to generate strong results. Description-centric processors still require extensive document pre-processing and translation into a cleansed environment, but do not require extensive processor tuning to generate robust results. An explanation-centric neurocognitive processor can generate human-like results on natural, unprocessed language documents with no processor tuning. However, in exploring human cognitive memory, it exposes the processor to the same human temporal sample biases not exhibited in accuracy or description centric processors.

CONFLICTS OF INTEREST STATEMENT

At the time of this writing, all authors have no conflicts of interest or any external funding from any entity.

REFERENCES

[1] Nigam, K., McCallum, A., Thrun, S., and Mitchell, T. (1998). Learning to classify text from labeled and unlabeled documents, Machine Learning, 39(2), 103-134.

[2] Roussinov, D., and Chen, H. (1999). Document clustering for electronic meetings: an experimental comparison of two techniques, Decision Support Systems, 27, 67-79.

[3] Yang, Y., and Liu, X. (1999). A re-examination of text categorization methods, Proceedings of the 22nd SIGIR, 42-49.

[4] Sebastiani, F. (2002). Machine learning in automated text categorization, ACM Computing Surveys, 34(1), 1-47.

[5] Jing, L., Huang, H., and Shi, H. (2002). Improved feature selection approach tfdif in text mining, Proceedings of the First International Conference on Machine Learning and Cybernetics, 944-946.

[6] Mittermayer, M. (2004). Forecasting intraday stock price trends with text mining techniques, In: Proceedings 37th Annual Hawaii International Conference on System Sciences (HICSS).

[7] Mooney, R., and Roy, L. (). Content based book recommending using learning for text categorization, In Proceedings of the Fifth ACM Conference on Digital Libraries, 195-204.

[8] Ikonomakis, M., Kotsiantis, S., and Tampakas, V. (2005). Text classification using machine learning techniques, WSEAS Transactions on Computers, 8(4), 966-974.

[9] Kim, H., Howland, P., and Park, H. (2005). Dimension reduction in text classification with support vector machines, Journal of Machine Learning Research, 6, 37–53.

[10] Lan, M., Tan, C., Low, H., and Sung, S., (2005). A comprehensive comparative study on term weighting schemes for text categorization with support vector machines, In Posters Proc. 14th International World Wide Web Conference.

[11] Zhang, W., Yoshida, T., and Tang, X. (2008). Text classification based on multi-word with support vector machine, Knowledge-Based Systems, 21, 879-886.

[12] Ko, Y., (2012). A study of term weighting schemes using class information for text classification, Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval, 1029-1030.

[13] Colace, F., De Santo, M., Greco, L., and Napoletano, P. (2014). Text classification using a few labeled examples, Computers in Human Behavior, 30, 689-697.

[14] Rumelhart, D., Hinton, G., & Williams, R. (1986). Learning internal representations by error propagation, Parallel distributed processing: explorations in the microstructure of cognition, vol. 1: foundations, MIT Press

[15] Rosenblatt, F. (1958). The perceptron: A probabalistic model for information storage and organization in the brain, Psychological Review, 65(6), 386-408.

[16] Cortes, C., & Vapnik, V. (1995). Support-vector networks, Machine Learning, 20, 273-297.

[17] Duda, R., Hart, P., & Stork, D. (2001). Pattern classification, Wiley-Interscience.

[18] Wong, C., and Versace, M. (2011). Context sensitivity with neural networks in financial decision processes, Global Journal of Business Research, 5(5), 27-43.

[19] Wong, C., and Versace, M. (2012). CARTMAP: a neural network method for automated feature selection in financial time series forecasting, Neural Computing and Applications, 21(5), 969-977.

[20] Bruel-Jungerman, E., Rampon, C., & Laroche, S. (2007). Adult hippocampal neurogenesis, synaptic plasticity and memory: Facts and hypotheses, Reviews in Neurosciences, 18(2), 93-114.

[21] Barnea, A. & Nottebaum, F., (1996). Recruitment and replacement of hippocampal neurons in young and adult chickadees: An addition to the theory of hippocampal learning, Proceedings of the National Academy of Sciences of the United States of America, 93(2), 714-718.

[22] Hall, J., Thomas, K., & Everitt, B. (2000). Rapid and selective induction of BDNF expression in the hippocampus during contextual learning, Nature Neuroscience, 3, 533-535.