PARASCRIPT®

Finding Key Information: Classification Techniques An Insider's Guide

Automated Document Classification

TABLE OF CONTENTS

INTRODUCTION:

The Classification Challenge

The human brain is the most advanced tool to organize information, recognize patterns, and distill critical insights. Our brains organize information adeptly in order to make quick, relevant decisions. Organizing and systematizing the spaces around us to function optimally is also known as classification.

Classification is critical for operating any successful business or organization. To effectively and efficiently manage information continues to become more challenging as the complexity and volume of data increases exponentially. The diversity of documents creates additional issues in accurately identifying the variety of documents created inside and outside of the organization. Organizations are looking to better classify and group documents in order to:

- Identify documents that are governed by regulations requiring proper retention
- Group documents to support business intelligence initiatives
- Sort and separate incoming documents to facilitate specific business processes
- Identify documents that are part of a legal action

Prevent certain documents or information contained within them from being sent or transmitted outside an organization.

Currently, some solutions that offer automated classification of documents exist, and they are not all created equal. One classification solution can be very different from the next. Different techniques may significantly affect the appropriateness or effectiveness of classification for any given project.

This eBook discusses the various document classification techniques, their strengths and weaknesses, and the resulting use cases appropriate for each of them.

CHAPTER 1:

Traditional Classification Techniques

Relevance and accuracy are the most important determinants in document classification success. While the “big data revolution” has brought classification into the spotlight, there is still a lot that is not generally understood.

The concepts of classification from a mathematical and information science perspective have been around for decades. Over the last forty years, many techniques have been introduced to tackle the challenges of relevance and accuracy.

As new business problems emerge, so too have new classification capabilities to specialize in solving these challenges. Classification techniques fall into two categories: text and visual classification.

TEXT CLASSIFICATION

Classification of text derives relevance to a given subject and essentially falls into two major camps: statistical and semantic. Most solutions are either one or the other, but can be used in tandem. Semantic classification derives relationships using the grammatical meaning of words.

Most often, semantic analysis extends analysis of words by adding synonyms and nearby words to attempt to derive the specific topic. Semantic-based classification requires a lot of initial work and upkeep, such as in the case of using slang or abbreviations.

Statistical classification, on the other hand, dispenses with deriving meaning by focusing more on the statistical relationships between words of a given volume. Using various statistical models, it is possible to have relevance without requiring grammatical elements that constantly must be managed.

The two methods can be used in tandem, but most classification approaches have a “core” of one or the other.

VISUAL CLASSIFICATION

While most of the attention is put on text-based capabilities, image-based classification using computer vision and pattern recognition offers very capable classification in its own right. This is because there is a lot of information regarding the visual components of a document, such as the presence of logos, tables, and layout, which can be used to group documents without text-based approaches. It's the same as sitting someone at a desk and asking them to group documents within a folder.

People often start by organizing like documents based upon visual cues alone—especially if they are not Subject Matter Experts (SMEs). Solutions using image-based classification vary by the granularity of the visual elements that the software can handle.

While some classification software take every visual component into consideration, others group by overall layout, ignoring logos or other picture-like elements. A benefit of visual classification is that it's very efficient in those cases where the layout of documents belonging to different classes differs significantly while the contents are similar.

Another type of visual classification deals more with actual imagery, such as a photo (e.g., a horse or landscape). This type of classifier works with more complex pictures beyond that of document-oriented visual analysis to determine a proper class based on factors such as geometry, color and other amorphous visual elements.

The benefits of both types of visual classifiers are that they do not rely upon processor-heavy OCR technologies.

CHAPTER 2:

Black box/User-centered Approach

Now that we have discussed text and visual categories of classification, it helps to discuss the various approaches to developing document classes. There are many ways, from labor-intensive approaches that involve SME reviews of document samples to highly automated software that derives groupings of documents, based upon identified features and attributes.

RULES-BASED CLASSIFICATION

Rules-based classification involves identifying specific characteristics of an object and then creating specific rules to govern if it is part of that class. For instance, the "apple" class can have rules that state that it is round, conic or oblong, red or green and sweet or tart. These are all explicit rules that can be used when analyzing the image of a fruit to determine if it can be placed into the "apple" class.

For documents, the process is similar. A person, typically a SME, will review a sample of target documents to determine if any specific characteristics can be used to place a document into a particular class. The SME might identify the presence of a company logo, certain data fields, or other "hints" that can be used for rules.

These rules can be very exact, such as "always look in the upper-right hand quadrant for a logo that matches this," or more flexible, such as "look for the word 'purchase order' anywhere on the document."

Rules can be effective for smaller scopes of document classes because SMEs can devote more time to creating the rules without developing many variances.

Rules typically use basic relations such as equals, greater, lesser, and, or, and other simple terms. Rules can be either simple with only one argument or complex using "and/or" to create more sophisticated outcomes.

SUPERVISED LEARNING-BASED CLASSIFICATION

Supervised classification uses machine learning to generate rules for how documents get classified. It is similar to rules-based methods in that a SME is still required to help the system learn different classes by providing examples for each class.

However, the person supervising the classification doesn't have to create explicit rules. This means that the system must be much more complex and must handle a much broader range of classes than an ordinary rules-based system. This is largely because the system creates a knowledge base of classification rules much faster than any person. As a result, classification capabilities cover a wider range of classes.

UNSUPERVISED CLASSIFICATION

Unsupervised classification is more commonly referred to as "clustering." It is the process of using machine learning to group samples into classes by their likeness. There are no SMEs necessary.

The benefits of unsupervised classification are that "a posteriori" activities do not require any previous experience or knowledge of the sample set. Grouping occurs based upon the facts that are identified within the sample group itself using probabilities without any "priming of the pump" with rules or any type of class definition.

Unsupervised classification will not typically have the same quality of results as learning-based classification due to the lack of any prior knowledge. What it lacks in accuracy, it makes up for in efficiency, especially where significant sample preparation effort can be mitigated.

CHAPTER 3:

The Best of All Classification Worlds

State-of-the-art document classification systems are based upon a combination of both visual and content approaches and can make use of both supervised and rules-based techniques.

Unsupervised classification is especially helpful during the initial stages of class preparation or discovery where there is a large set of target documents without much organization.

Classification and extraction capabilities support multiple document types and handle various scenarios including extraction data from structured and semi-structured forms. Unstructured documents can include a variety of data, such as handwritten forms or machine print, the presence of imagery or logos, script, tables or other important characteristics.

For reliable classification, it is necessary to analyze the content and the layout of a document. The most useful systems also offer auto-classification capabilities that use the:

- Results of the content;
- Layout analysis; and
- Intelligently combine the outcomes for higher accuracy.

The result is a class that uses all of the data in the document to determine a single class.

DEVELOPING THE KNOWLEDGE BASE

Classification starts with Learning that uses a predefined knowledge base. In the case of Parascript classification, the knowledge base can be a simple hierarchical folder structure consisting of a number of subfolders.

Each subfolder represents a class of documents, has a class name, and contains images of documents belonging to the corresponding class. As soon as the system is trained, it can reliably classify documents related to classes included in the knowledge base.

Developing all the predefined categories up-front is sometimes unfeasible to create a robust, predefined knowledge base. In such cases, unsupervised learning is the approach that makes the classification task feasible.

The most advanced systems offer technology that may work with the full spectrum of diverse documents automatically dividing them into categories based on content or layout similarity without any pre-knowledge provided by human operators. Unsupervised systems are not provided any training examples at all, so the learning process attempts to find appropriate "categories," analyzing the documents in their entirety.

This analysis process is referred to as "clustering." The absence of training example preparation makes the unsupervised paradigm very appealing. Moreover, the results of clustering algorithms are data driven, hence more 'natural' and better suited to the underlying structure of the data. This advantage is also the major drawback of clustering. Without a possibility to tell the machine what to do (as in supervised classification), it is difficult to judge conclusively the quality of clustering. Clustering results are further analyzed and refined manually.

Automatically created clusters may be modified (divided, merged, or documents moved from one cluster to another) to be converted in a knowledge database and used for fine-tuning a classifier. This approach uses the benefits of both techniques. It allows the system to deal with larger data sets and to model the inherent structure of the data. Simultaneously, it improves the accuracy and controllability of results in those cases where categories can be predefined.

CLASSIFICATION TECHNOLOGIES

Both classification and clustering involve pattern recognition technology. Content classification employs text features of documents. Layout classification mainly uses geometrical features of documents, (e.g., structure of text lines, frames and titles) and they go beyond that.

Document processing technology has come a long way and can solve a lot of problems from converting paper documents into a digital image to reading machine printed pages (OCR) or extracting data from structured and semi-structured documents filled out with machine-print or handwriting. However, classification of multiple types of documents is still challenging, requires a lot of effort, and truly sophisticated technological capabilities.

At first glance, using rules based upon keyword search appears to solve the problem. For example, to classify invoices, simply extract the page of text and look for the word "invoice."

However, this approach is not sufficiently reliable as a classification solution. One keyword may be missed or not found. In the case of OCR, the system may fail to read the keyword. As a result, the document won't be associated with the right class. To solve the problem of classification, it is necessary to treat the document holistically, reading and relying not just on one word, but on all words in the document and to distinguish between classes, using calculated statistics.

Currently, multiple methods and algorithms are used to build a document classifier: the decision tree classifier, the naive Bayes classifier, the nearest neighbor classifier, support vector machines (SVM), artificial neural networks, Linear Discriminant Analysis (LDA) algorithm, and many others including proprietary methods.

- Decision tree induction is the learning of a decision tree from class-labeled training tuples.
- Bayesian classifiers are statistical classifiers and are based on Bayes theorem.
- SVM has its roots in statistical learning theory and has shown promising empirical results in many practical applications, from handwritten digit recognition to text categorization.
- The Max Entropy classifier is a probabilistic classifier that belongs to the class of exponential models and from all the models that fit the training data, selects the one with the largest entropy.
- An artificial neural network is a computational model based on biological neural networks. Some of them are more efficient for content-based classification. Others are optimal for layout-based classification.

Each method has its strengths and drawbacks and may be more or less efficient depending on the classification problem that it must solve. Strengths and weaknesses of these methods are summarized in the table on the following pages.

Method	Strengths	Weaknesses
Decision Trees	Easy to understand and explain Perform classification without requiring much computation Transparent results, can be interpreted as explicit If-Then rules Based on few assumptions, works well even with missing data and outliers Provide a clear indication of which fields are important for prediction or classification	Accurate results require very large databases Are prone to errors in classification problems with many classes and relatively small number of training examples Can be computationally expensive to train
Naive Bayes Classifier	Fast to train , fast to classify Not sensitive to irrelevant features Handles real and discrete data Handles streaming data well	Assumes independence of features
Nearest Neighbor Classifier	Simple to implement and use Comprehensible – easy to explain prediction Performs well where there is a large training database, or many combinations of predictor variables	Need a lot of space to store all examples Long computational times May show worse results on data other than the training set

Method	Strengths	Weaknesses
Support Vector Machines (SVM)	Training is relatively easy Scales relatively well to high dimensional data Tradeoff between classifier complexity and error can be controlled explicitly Nontraditional data like strings and trees can be used as input to SVM, instead of feature vectors	The more samples in the training set, the more complex and slow the classification process Need a "good" kernel function, i.e., similarity defined by the kernel function should have good correlation with the type of the problem
Artificial Neural Networks	Highly flexible Capture complex relationships between inputs and outputs	Learns too well in the training data session but generates inferior results in case of out of sample session The construction of the NN model can be a time-consuming process Limited insight into underlying relationships Requires a lot of examples for training
LDA Algorithm	Small model size Fast classification speed Computational efficiency	Not applicable to complex tasks

CHAPTER 4:

New Classification Approach

While many of the techniques used by Parascript document classification are proprietary, we would like to describe some important techniques and the rationale behind them. One of the mechanisms implemented in Parascript's Content classifier is a well-known naive Bayes classifier, based on applying Bayes' theorem with strong (naive) independence assumptions between the features. A naive Bayes classifier assumes that the presence (or absence) of a particular feature of a class is unrelated to the presence (or absence) of any other feature.

Usually we deal with a priori probability; when the class of an object is known, there is a certain probability that it has a given set of features. For example, the object is an orange so we expect that it is round with a bright reddish-yellow rind and is about 4" in diameter. To solve a classification problem requires the opposite task: knowing a set of features, the class of the object must be determined.

Bayes' theorem allows us to find the a posteriori probability of an event, based on the characteristics that we observe, (e.g., a fruit may be considered to be an orange if it is round, has bright reddish-yellow rind and is about 4" in diameter). Even if these features depend on each other or upon the existence of the other features, a naive Bayes classifier considers all of these properties to independently contribute to the probability that this fruit is an orange.

In text classification, different words in the text on the same topic appear independently. Imagine each topic as a bag of words which contains words that are related to this topic. Each word may be met a few times, and different words occur a different number of times. Every time a text in a certain topic is generated, the word is extracted and used in the text.

The reality is much more complex. Nevertheless, the naive Bayes classifier frequently outperforms other methods and proves to be very useful in solving real-life classification problems. One of the reasons is that although words in reality are dependent on each other, this dependency is the same for different classes. The grammar and semantic dependence between words are the same despite the topic of the document. Therefore, it may be reasonable to ignore dependencies when probability is estimated.

The Naive Bayes classifier has many other advantages. It is less computationally intensive than other methods. Unlike neural networks, for example, it may work with a small amount of training data to estimate the parameters necessary for classification. This characteristic effectively reduces the amount of preparatory work required to achieve reasonable classification performance; and allows naive Bayes classifier to be trained rapidly and efficiently.

Another important benefit of this technique is that its efficiency does not depend on the size of the document. With proper normalization, correct classification is possible when document sizes vary significantly—not only in different classes—but within the same class as well.

For layout-based classification, the K-nearest neighbors algorithm is often used. The basic principle of the method of nearest neighbors is that the object is assigned to the class, which is the most common among the neighbors of a given element.

Depending on the type of needs and data, either layout- or contents-based classifier works better. However, combination of classifiers built on different approaches may produce the best results. This approach combines trained classifiers and performs classification by taking a vote on the predictions made by each of them and results dramatically depend on the quality of voting algorithms.

Advanced proprietary methods and know-how allow the efficient combination of classifiers based on orthogonal approaches, which makes the difference in the efficiency of this method. And yet, many real life applications are dealing with data that contain inherent uncertainty and building a reliable classifier for them remains a great challenge.

Any additional specifying information may help and significantly improve the efficiency of an automatic classification process in such cases. This information can be provided in some cases in the form of rules written by people. Classifiers that involve rules are called rule-based.

A rule-based classifier is a technique for classifying data using a collection of ”if ... then ...” rules that show the relationship between the attributes of a document and the class to which it belongs. Rule-based classification algorithms have a number of benefits:

- Rule sets are relatively easy for people to understand and classification systems that involve rules outperform other classification methods on many problems.

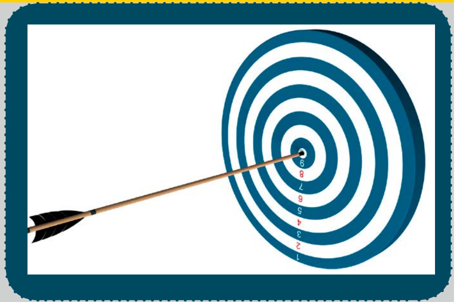
- Rule-based classification also offers flexibility to efficiently involve the output received from other advanced document analysis and data extraction methods and technologies into classification process.
- The results of highly accurate handwriting detection, signature presence detection, signature verification, or logo-matching techniques can be included in classification process via rules.

Another important problem that is crucial to accurate classification relates to making the right decision as to whether the document that has to be associated with a certain class is similar enough to the documents related to this class. Documents in some classes are more similar to the document in question; and in others, they are less similar to the document in question.

There is always a certain value provided in the answer that characterizes the "response" of each class to the document in question or, in other words, the similarity between the documents in a certain class and the document in question. It is important to guarantee that if a document does not belong to either class, the "response" of each class to this document is below a certain threshold. Rich experience accumulated by Parascript solving the recognition tasks allows the company to provide a solution that minimizes this type of error.

CHAPTER 5:

Document Classification Results



In the end, it is all about the results that come from classification of any kind. While many classification projects focus on the accuracy of the classification itself, another key factor is the error rate. With any machine learning application, the system will make attempts at arriving at the correct answer and it will also make errors. Error rates measure the documents that were classified but done so incorrectly. In many systems, the only way to find these "false positives" is to review the entire results of classification. Mature classification technologies try to deal with this problem through some sort of scoring mechanism that is used to indicate when the classification result for any document is suspect and should be reviewed.

With machine learning, confidence and error rates apply not only to classification but also to data location and extraction within a document. As an illustration, for location and extraction of the date field, you can see the various answers from the system along with a score. Scores are used to indicate the likelihood of any particular answer being the correct answer; the larger the score, the more likely the answer is correct – judged by the system as is shown in Figure 1. Scoring. Parascript software analyzes both the entire field as well as the individual characters, which you can see in Figure 2. Individual Character Analysis.

Figure 1. Scoring

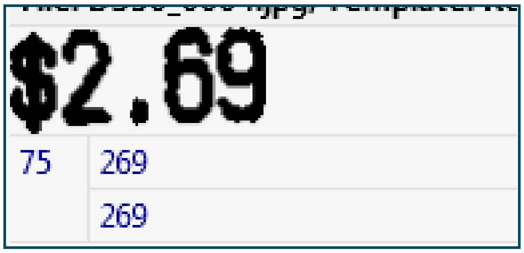
Answer	Score
09/28/2014	960
09/28/2014	960
09/28/2011	440
09/28/2012	440
09/28/2013	440

Figure 2. Individual Character Analysis

Symbol	Confidence	Coordinates
9	100	0 2 19 36
/	100	22 0 16 39
2	100	40 2 25 35
8	100	53 2 26 35
/	100	80 0 18 39

Field-level and character-level scores are used to arrive at a synthesized score for the entire answer. As we see in Figure 3. Confidence Level, using a total field on a receipt, the system has scored the answer at a confidence level of 75.

Figure 3. Confidence Level - 75



$2.69

75 269

269

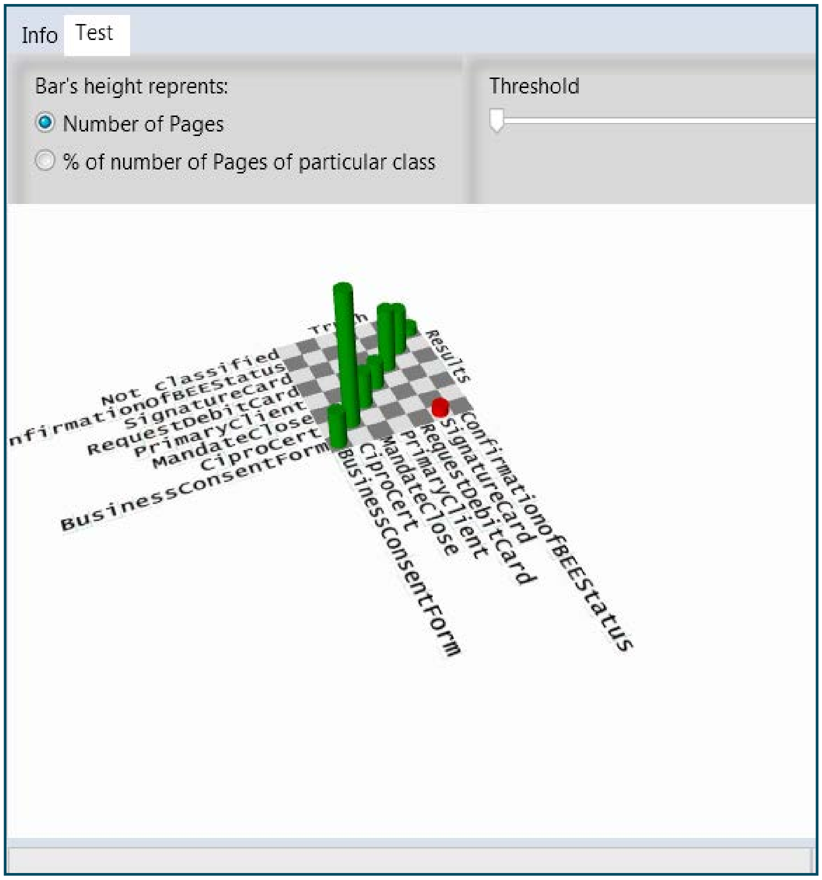
Through proper analysis and fine-tuning using a larger sample of documents, it is possible to arrive at a particular confidence score for document classes and the individual data fields within each document that represents the threshold between highly accurate classification results and those that require further analysis.

With regard to error rates, it is important to understand the instances where the system classifies a document incorrectly but provides a confidence score that meets a selected threshold that would indicate the class assignment to be correct. These instances are called "false positives" and can have significant effects in production as these results are treated as correct without any validation.

With error rates, there are two primary components: 1) the rate of acceptance and 2) the associated error rate. Rate of acceptance is the percentage of results that are passed through the system as accurate and that do not require manual review or auditing. The error rate is the percentage of those accepted results that are erroneous. The objective is to have a high-enough acceptance rate which results in lower review costs with the lowest error which means that the results passed through are as accurate as possible.

Figure 4. Error Rates with Thresholds shows an example of error rates at a variety of thresholds. Using sample sets of documents along with what are called "truth data" that provides the actual answers for the correct class of each document allows an analyst with the ability to view a range of error rates based upon their own business requirements.

Figure 4. Error Rates with Thresholds



Info Test

Bar's height reprents:

● Number of Pages

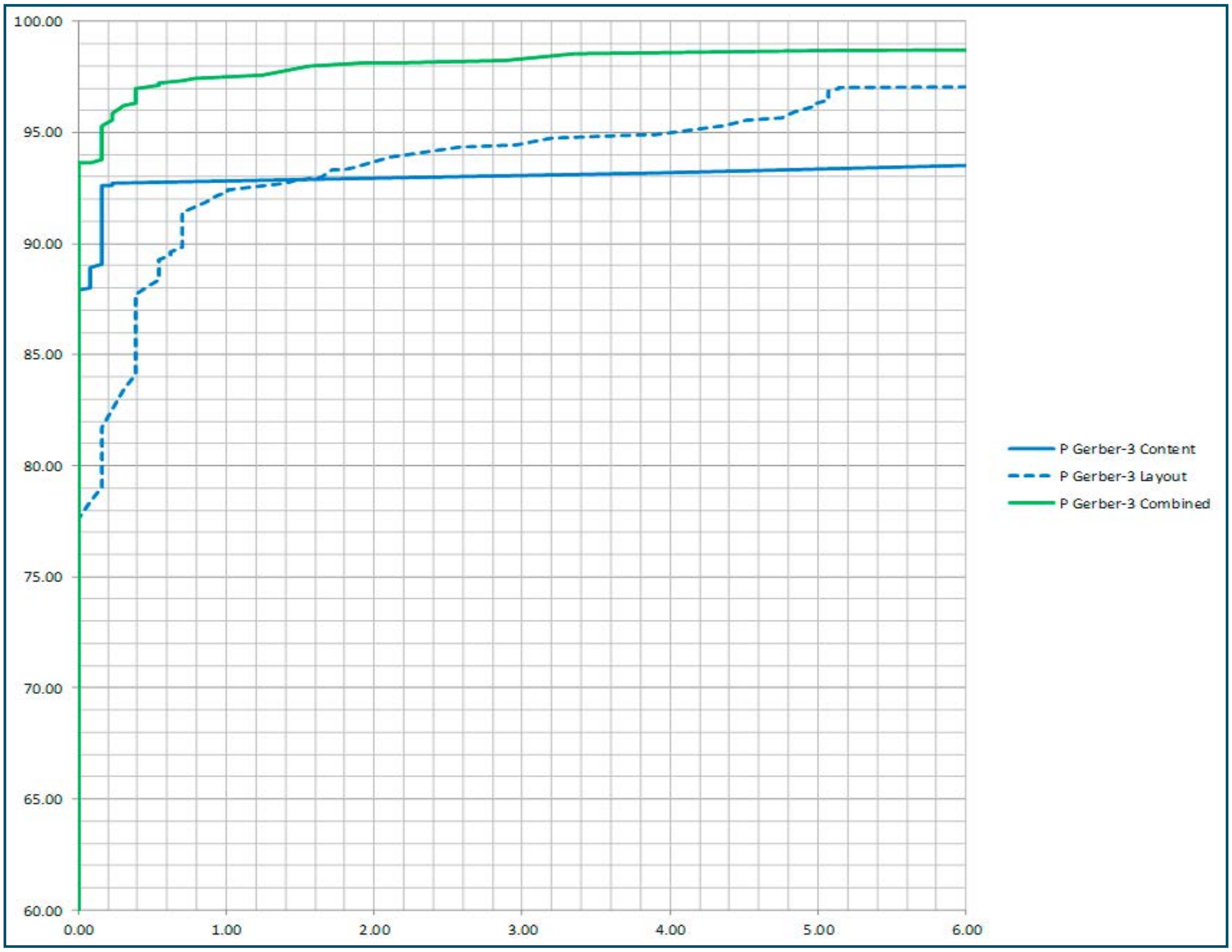
○ % of number of Pages of particular class

Threshold

It is always important to run tests using sample classification sets with known classes in order to understand the error rates of the system. With Parascript document classification, running these tests is easy, and the results are displayed visually as shown in *Figure 5. Sample Results*. In this example, we see that the system is set-up to identify seven different document classes associated with a bank project. Once documents are run through the system, users can view the results across different confidence levels and view the error rates which are displayed as red bars within the graph. By simply sliding the confidence level back and forth, the user can view the resulting accuracy and error of each document class.

Parascript Classification is designed to have the highest accuracy with the lowest error in order to produce truly relevant classification results. In internal measurements against a large sample set, you can see that using our unique combination of both visual and content classification, acceptance rates is in the 90 percent range with practically zero errors and can achieve acceptance rates of almost 100 percent with only single digit error rates. Error/Acceptance curves for visual and content classification separately can still achieve rates of single digit error with 90 percent acceptance.

Figure 5. Sample Results



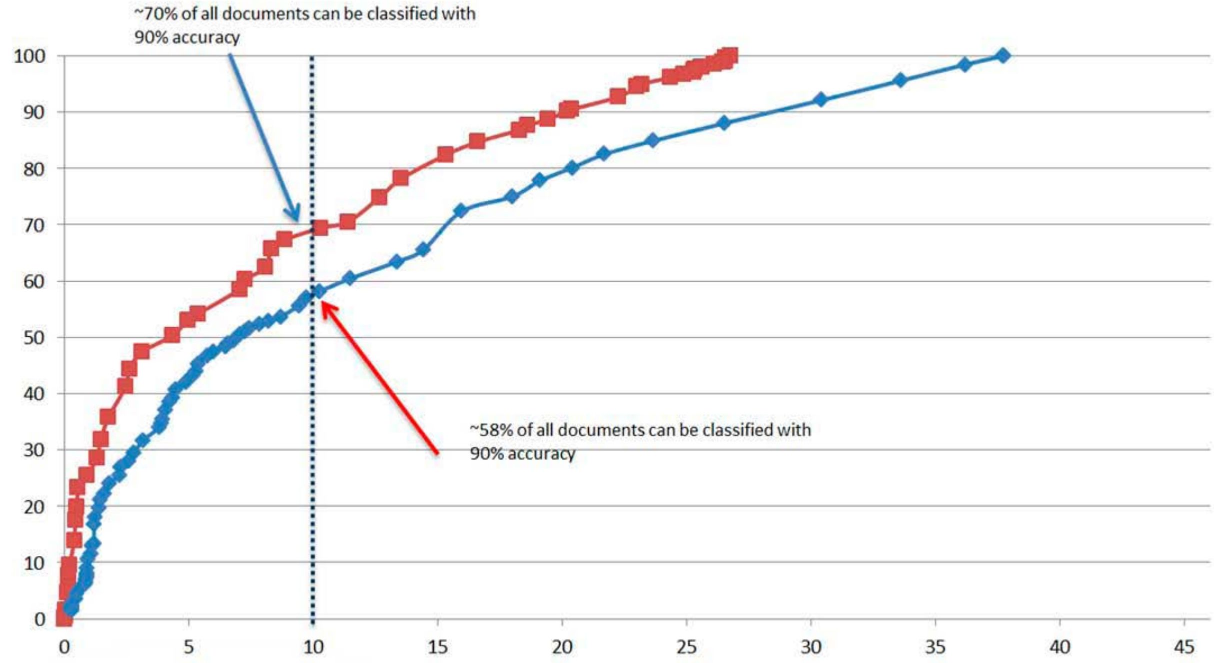
CLASSIFICATION OF COMPLEX DOCUMENTS

There are many examples of simple document classification where the task is only to sort and identify a relatively small number of potential document types. When it comes to more complex document classification needs, such as in mortgage or auto loan processing, requirements can become much more complex including the need to take an individual multi-page file and then classify and separate it into individual documents. Potential document classes can number over 1000.

In these cases, training document classifiers becomes more complex due to the potential for each new document class to "confuse" the classifiers. For instance, we train a document classifier on 100 different classes, there is some potential of the classifier to confuse one document that belongs to a particular class with another document class. This is caused by similar "features" between one class and another. The result is a misclassified document. When there are only 20 potential classes, confusion is unlikely to be a problem. As we increase the number of potential classes beyond 30 or 40, potential for error can increase significantly. When the number of classes moves beyond 100, there can be substantial error.

The only recognized way to reduce potential error is to test and tune the system. Classification results must be reviewed with analysis primarily focusing on misclassified documents. Each error must be analyzed for potential confusion (e.g., understand similarities and differences with misclassified documents) to determine if the system needs to be retrained with new samples or with new information. This process must be done each time a new document class is added or the system performance will degrade.

Figure 6. 20% Improvement over Competitor Mortgage Classification Solutions



HOW TO OVERCOME THE COMPLEXITY CHALLENGE

As can be imagined, this process of testing, review and tuning can take several hundred hours to optimize performance. To tackle this challenge head-on, Parascript created a new type of classifier that builds-in the testing, analysis and tuning in order to produce optimized document classification regardless of the number of classes involved.

Parascript document classification effectively automates 100s of hours of manual effort and produces the industry's highest performance as is evidenced by the below comparison with an industry leading solution.

In Figure 6, Parascript technology achieves a 20% improvement over a competing mortgage classification solution. Parascript document classification simultaneously delivers higher performance and lower configuration costs.

CONCLUSION

While classification technology has been around for quite some time, the combination of increased computing power and new novel combinations of several types of classification means that the ability to automate typically costly document organization and taxonomies is more achievable and approachable today by any organization, large or small.

Using a careful blend of visual, content, and rules allows businesses to approach once-costly projects with a cost-effective and reliable means to go from content chaos to content bliss. Parascript Classification offers this capability along with an almost obsessive focus on high accuracy combined with low error rates to provide truly relevant classification results.

Finding Key Information: Classification Techniques An Insider's Guide

Find out how we can meet your document classification needs.

Contact us today at:

888.225.0169
info@parascript.com

PARASCRIPT®

www.parascript.com

Parascript Management, Inc. | 6273 Monarch Park Place | Longmont, CO 80503 USA