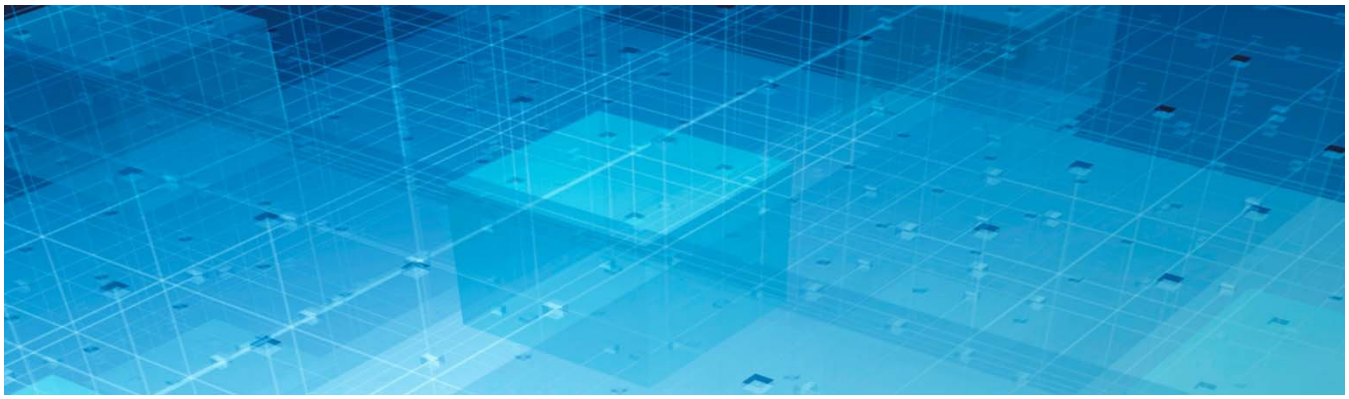
paradatec



Analysis of Document Recognition Methodologies

Overview

The objective of this paper is to highlight the differences in methodology between the document recognition technology offered by Paradatec and that offered by other vendors. There is no intention to identify specific technologies by name. We hope that the comparison will allow the reader to make more informed decisions about the environments that may be more suitable for each methodology.

High-Level Approach

The most fundamental difference in approach between Paradatec and the majority of other "advanced" document recognition technologies is that Paradatec treats variable layout documents as *unstructured* documents whereas most other prominent solutions treat them more as *semi-structured* documents. To illustrate the difference in methodology let us consider a customer wishing to process pay stub documents. A Paradatec implementation would typically be deployed with one set of completely generic rules designed to encompass all variations of the "Pay Stub" document type from any company. The software is capable of being configured to perform conditional processing for specific exceptions to our generic rules (per-layout exceptions) but it is not typically necessary to do this.

Most other modern advanced document recognition technologies treat document variations as semi-structured documents. These solutions typically either:

a) Remember as many of the variations as is practical and process each variation with layout-specific templates for processing

Or

b) Apply a mix of layout-specific processing and some generic processing (usually the higher incidence layouts are processed with layout-specific or templated processing)

The obvious advantage of our generic approach is that, as new layouts appear or existing layouts change, the software is better equipped to deal with these new variations. This advantage was recently validated at a Paradatec account with over 35,000 layout variations where rules had been in place for five years with not a single modification. An audit of this client's processes after five years revealed an identical automation rate to that when the system was first deployed. This outcome was observed despite the fact that a significant portion of the originally dominant layouts had been transferred to EDI processing and were therefore bypassing the system now.

Document Classification Methodologies

There are three typical methodologies applied to document classification. We shall provide a high-level overview of each in turn:

1. Methodology, Judicious Learning

Paradatec Document Classification, based on our "Judicious Learning" philosophy, is distinct from most other classification technologies in that a fundamental design tenet says that anything learned must be thoroughly tested in a lower environment prior to being promoted to production. The system is licensed to allow virtually unlimited regression tests to be performed, even daily if required.

Our solution reads all text on every page, applying a combination of artificial intelligence (AI) and user-guided logic to derive context from the text, just as a human does. The learning process occurs in a lower level, non-production environment where our AI engine is taught about context and decisioning. As an example, "learning" in the mortgage industry consists of running full loans of pre-classified pages through the software, with the AI engine reading all text on each page in less than half a second on current generation processor technology. The system then applies a set of AI and user-guided rules to determine which document type the page belongs to. Once the Paradatec system is trained, regression testing takes place on a set of up to one thousand loans to verify that the latest build is an improvement in every way from the prior. Then, and only then, will the system be promoted to production.

The reason that most other methodologies employ alternate approaches (like only reading specific zones on a page) is that reading the entire page can take five seconds or more with other products. Paradatec's technology was built from the beginning with the ability to read every page in its entirety and is highly optimized for that task. This Paradatec recognition engine has been in production since 1997 and has processed billions of pages in that time. Many other solutions began their lives as zone-based OCR products and *perhaps* they've developed the ability to read the entire page later in the product's lifecycle, but many have still not achieved that mark.

How Classification Works – The journey from Test to Production

The Paradatec document types, (e.g. "Mortgage Note", "Certificate of Title", and "Appraisal"), each encompass all possible layouts of those document types. This is different from most other solutions in that unique templates for each document layout variation are not required, greatly reducing the level of administrative support required. Artificial intelligence in the self-learning Paradatec classifier then analyzes every word and phrase on every page in a PDF or TIFF multi-image file. This artificial intelligence identifies either:

a) Many individual words or phrases that are unique to a particular page

And/Or

b) Words and phrases that are not necessarily unique but, when found in a certain relative spatial position, indicate uniqueness. For example, in the mortgage world, a "Mortgage Note" document may have words/phrases such as "Note", "Promise To Pay", "Interest Rate", and "monthly payment will be" but not necessarily all of these. The rules may determine a higher certainty that the document is, in fact, a Mortgage Note document if any of the first three terms occurs lexically prior to the term "copy of this disclosure" (since this is usually near the bottom of the document).

How Paradatec Classification Rules are Created

Paradatec rules can be created in the following ways:

- Automatically learned from a set of pre-classified document examples
- Manually configured rules
- or by a combination of both of these methods

The Self-Learning Object, when given a set of pre-classified documents, will:

- Read and capture every word on each page – and its location - from every document type in less than one second per image
    - The location metadata is a key to understanding context of relevant phrases and data elements
- Determine the statistically unique collections of words and phrases that accurately identify each document

Document Boundary Detection within a Paradatec Project

Paradatec software is unique in that it eliminates the need for document *separator sheets* to be added between multi-page documents. Our Document Boundary Detection Object differentiates between the first page of a multi-page document and subsequent "Following Pages" via AI-based analysis of the text on these pages. There are several complementary techniques that allow this process to work accurately:

a) Identification of page numbers from sequential numbered pages
b) Identification of unique text on following pages
c) Comparison of meta-data from page to page

Item a) above is fairly self-explanatory, in that a page with the text "Page 2/4" will be classified as a "Following Page". Other than environments with fax-only input, numbered pages are not always common.

Item b) above is also fairly straightforward, in that the software learns the text signatures of following pages.

Item c) is fairly unique and is valuable in many different environments. The software remembers information from page to page to help to determine if the current page is a continuation of the last page processed. For example, consider a document with a Roman numbered list. Perhaps items i) and ii) are on page one and items iii) and iv) are on page 2. The system can recognize that the numbered list continues across pages and can associate the pages together. Similarly the document boundary detection object can compare margins, fonts, headers and footers across adjacent pages in a "blob" and automatically infer where the document breaks are, without detailed prior knowledge of the document.

Error rates measured by downstream human audit of document boundaries show that the combination of the three techniques described above yield error rates significantly below 1%.

Other Methodologies in Advanced Document Recognition

2. Methodology, Visual Classification

Typically, this methodology is deployed if the document recognition system does not have the ability to read an entire page of text in less than a few seconds per page, or if the document content is more graphically oriented rather than textual. With this approach, an *image analysis* approach is used to identify document types. This works as follows:

A set of pre-classified reference documents is processed. The system attempts to differentiate between document type A and document type B largely by examining the distribution of ink on samples of each document type. This is like a thumbprint analysis, i.e. a graphical signature of each document type that is learned and remembered. If the system sees substantial variation in the image signature from documents that were all pre-classified as document type A, it will suggest to an operator that these be split into more document types where each document type has a consistent shape (or image fingerprint). This works fine for environments where there are a relatively small number of structured forms. This approach can also be applied to environments where there are a very few document types but quite a few layout variations within a document type. An example might be an accounts payable operation that processes invoices and credit memos from only thirty main vendors. In this scenario, the system will break down all of the invoices into thirty groups (one per vendor) since each vendor's invoices have a unique image fingerprint. Visual classification sometimes requires human insertion of separator sheets between multi-page documents.

This image-based methodology works well for fewer document variations but can start to produce more inconsistent results when the number of document variations is greater than about 100 possibilities. The practical solution commonly employed in the case of high variability is a *waterfall* process. This is a hybrid approach, wherein the image signature step described above is applied to the most prevalent document types (or document variations, since this is the fastest approach), and then textual analysis is selectively used to distinguish the other, more difficult document types or variations. During this step, as a shortcut, textual analysis is often performed only on specific zones of a page due to the performance limitations of most OCR engines. Visual Classification typically relies on third party OCR components rather than functioning as a homogeneous solution. In general, for pages that can be identified by visual means only, i.e. without OCR, the system is very fast, possibly processing ten pages per second per CPU core.

For data extraction, this methodology remembers the layout of the document variation and is able to perform a layout-specific OCR of the areas on the page from where data is to be extracted. Anchors are used and relative physical offsets are remembered from these anchors to locate index data. Anchors are often implemented as graphical anchors (rather than text) in keeping with the overall philosophy of image-based rather than text-based analysis. Table data extraction is possible with this methodology, but success depends upon the complexity of the tables.

Advantages of this methodology include:

- Performance (for the images able to be processed by the image signature method)
- Training time (it is relatively simple for the system to learn a small number of document types from image signature analysis).

Disadvantages of this methodology include:

- The layout-specific configurations needed for each document variation can take a long time to set up if the number of document variations/types is high.
- These layout-specific configurations need to change if the layout of a document ever changes.
- The graphical signature approach tends to be less reliable with more than one hundred document variations/types to compare. This can affect accuracy in some cases.
- The time to process images tends to be linearly related to the number of document variations/types.
- This approach presents challenges when attempting to detect document boundaries for multiple page documents and does not provide an ability to extract data from the documents once identified.

3. Methodology, Learning in Production

This methodology does not have the advantage of a sub-second textual analysis solution but it does use text analysis as part of its document classification and data extraction solution. In general, the system is a mix of preconfigured rules, a learned knowledgebase and layout-specific configurations. The rules are configured through a GUI, but more complex operations require scripting and development effort. The technology is often configured for mailroom and Accounts Payable environments.

Learning in Production works as follows in an Accounts Payable environment:

During initial configuration the system augments the accounts payable rules that come with the system by learning. This learning is achieved by running real production data through the system for verification. The system attempts to learn from the document classification decisions made by the verification operator. For example, suppose the operator identifies an invoice as being from vendor XYZ. The system will keep a record of the incidence of all the words on the invoice with the record it associates with this vendor. Furthermore, it will keep a record of the number of occurrences of each word. This is the signature that this methodology uses to identify document variations. The signatures are learned during production. Up to four samples of each layout are used to reinforce the rule learned for that layout. Learning in Production may require insertion of separator sheets between multi-page documents depending on how clear the document boundaries are.

Generic Learning involves passing many invoices through validation. The data is identified manually and the invoices are marked for generic learning. An off-line process then searches through a full-page OCR read of the document looking for relatively unique words associated with the template and remembers the unique words. Manual review and training of the generic knowledge base is need to complete the learning process, typically requiring programming-type personnel to complete. Generic learning is built around the concept of Support Vector Machine (SVM). The principle is that the SVM technique is inherently more reliable than other types of learning when the variation of layouts per document is very low. However, for markets with large variations in document layouts, such as the mortgage market, the SVM concept is not optimal.

Specific Learning is an in-line process that is very layout specific. In this approach, a verification operator marks an invoice for specific learning by the system. The system then learns the location of the relevant data on that page based on information from the verification operator. These template-like improvements are then available in the next batch processed (in production). The assumption in this approach is that the verification operator always identifies the correct value, since this approach doesn't support a regression testing cycle. The downside to this is often that verification operator errors and statistical outliers can become training elements that skew an otherwise optimized system.

One of the advantages of this methodology is:

- In-production learning allows rapid use of layout specific information. This advantage is also a disadvantage. Many higher-volume sites require regression testing prior to promotion of any configuration change into production. This methodology is based on a belief that this is not necessary.

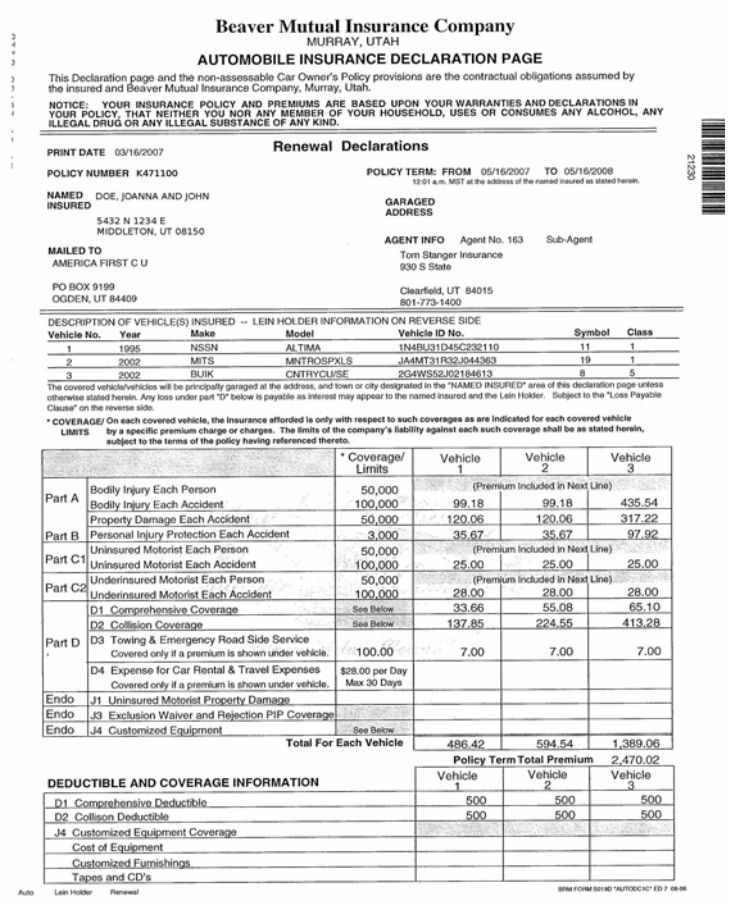
Disadvantages of this methodology include:

- As the system adds layout-specific templates, the system gets proportionately slower
- The system only is implemented from within a proprietary capture platform from the same source (not an open plug-in)
- Separator sheets between multi-page documents are required
- Mistakes will be made if layouts change
- Focused primarily on mailroom and accounts payable functions

Paradatec Data Extraction

Paradatec's proprietary AI text analysis application, PROSAR-AIDA, is not trained to read the layout of a specific document, but rather to *discover* "on-the-fly" the attributes of a document by reading the entire document in context and then *discovering* the appropriate information (just as a human would).

Consider the Insurance Declaration document to the right. The software is trained to know that this type of document typically has a Policy Number and Term as well as a series of tables relating to coverage. The software will identify, say, the Policy Number and Term by reading all of the text on the page in order to identify the labels and data relating to the information needed, just as a human would. In the case of the Policy Number, the software will be aware of a range of synonyms of the label POLICY NUMBER e.g. Policy Num, Pol. #, …



Beaver Mutual Insurance Company
MURRAY, UTAH

AUTOMOBILE INSURANCE DECLARATION PAGE

This Declaration page and the non-assessable Car Owner's Policy provisions are the contractual obligations assumed by the insured and Beaver Mutual Insurance Company, Murray, Utah.

NOTICE: YOUR INSURANCE POLICY AND PREMIUMS ARE BASED UPON YOUR WARRANTIES AND DECLARATIONS IN YOUR POLICY. THAT NEITHER YOU NOR ANY MEMBER OF YOUR HOUSEHOLD, USES OR CONSUMES ANY ALCOHOL, ANY ILLEGAL DRUG OR ANY ILLEGAL SUBSTANCE OF ANY KIND.

Renewal Declarations

PRINT DATE 03/16/2007

POLICY NUMBER K471100

POLICY TERM: FROM 05/16/2007 TO 05/16/2008
12:01 a.m. MST at the address of the named insured as stated herein.

NAMED INSURED DOE, JOANNA AND JOHN
5432 N 1234 E
MIDDLETON, UT 08150

GARAGED ADDRESS

MAILED TO
AMERICA FIRST C U
PO BOX 9199
OGDEN, UT 84409

AGENT INFO Agent No. 163 Sub-Agent
Tom Stanger Insurance
930 S State
Clearfield, UT 84015
801-773-1400

DESCRIPTION OF VEHICLE(S) INSURED -- LEIN HOLDER INFORMATION ON REVERSE SIDE

Vehicle No.	Year	Make	Model	Vehicle ID No.	Symbol	Class
1	1995	NSSN	ALTIMA	1N4BU31D45C232110	11	1
2	2002	MITS	MNTROSPXLS	JA4MT31R32J044363	19	1
3	2002	BUIK	CNTRYCU/SE	2G4WS52J021184613	8	5

The covered vehicle/vehicles will be principally garaged at the address, and town or city designated in the "NAMED INSURED" area of this declaration page unless otherwise stated herein. Any loss under part "D" below is payable as interest may appear to the named insured and the Lein Holder. Subject to the "Loss Payable Clause" on the reverse side.

* COVERAGE/LIMITS On each covered vehicle, the insurance afforded is only with respect to such coverages as are indicated for each covered vehicle by a specific premium charge or charges. The limits of the company's liability against each such coverage shall be as stated herein, subject to the terms of the policy having referenced thereto.

		* Coverage/ Limits	Vehicle 1	Vehicle 2	Vehicle 3
Part A	Bodily Injury Each Person	50,000	(Premium Included in Next Line)		
	Bodily Injury Each Accident	100,000	99.18	99.18	435.54
	Property Damage Each Accident	50,000	120.06	120.06	317.22
Part B	Personal Injury Protection Each Accident	3,000	35.67	35.67	97.92
Part C1	Uninsured Motorist Each Person	50,000	(Premium Included in Next Line)		
	Uninsured Motorist Each Accident	100,000	25.00	25.00	25.00
Part C2	Underinsured Motorist Each Person	50,000	(Premium Included in Next Line)		
	Underinsured Motorist Each Accident	100,000	28.00	28.00	28.00
Part D	D1 Comprehensive Coverage	See Below	33.66	55.08	65.10
	D2 Collision Coverage	See Below	137.85	224.55	413.28
	D3 Towing & Emergency Road Side Service Covered only if a premium is shown under vehicle.	100.00	7.00	7.00	7.00
	D4 Expense for Car Rental & Travel Expenses Covered only if a premium is shown under vehicle.	$28.00 per Day Max 30 Days			
Endo	J1 Uninsured Motorist Property Damage				
Endo	J3 Exclusion Waiver and Rejection PIP Coverage				
Endo	J4 Customized Equipment	See Below			
		Total For Each Vehicle	486.42	594.54	1,389.06
			Policy Term Total Premium		2,470.02

DEDUCTIBLE AND COVERAGE INFORMATION	Vehicle 1	Vehicle 2	Vehicle 3
D1 Comprehensive Deductible	500	500	500
D2 Collison Deductible	500	500	500
J4 Customized Equipment Coverage			
Cost of Equipment			
Customized Furnishings			
Tapes and CD's			

Auto Lein Holder Renewal

Table Data Extraction

The Paradatec table module can automatically detect tabular data and export that data in a normalized and consistent format, despite the table variations. Table detection is set up via a GUI and has a number of important features such as:

- Supports multiple tables per page.
- Understands tables that span pages.
- Math operations are used to help determine which column is which (e.g. for invoice line items, the software knows that [Unit Price – Discount] x Quantity = Total Cost and this information helps to determine on-the-fly (without a template) which columns are which).
- De-interlaced columns (e.g. on a medical Explanation of Benefits document, a Date of Service might be physically placed on the same column as, say, the Billed Amount. The table module will separate these into two columns).
- Unstructured content columns (e.g. description columns) can be accurately identified.
- Can split a column into two (e.g. we can separate a part number from somewhere in the middle of an item description and return it as a separate column).
- Wrapped entries in columns of the type above can be unwrapped to one line (see Part D in the image on the prior page)
- Use of OMR (checkboxes or bubbles) objects in tables
- Line items that are wrapped over more than one line (our solution will unwrap and deliver a single line in the XML output)

It is important to understand that all of these features are *inherent to the* Paradatec *table module* and require *no programming or scripting*. Everything regarding setup is done via a GUI and NOT on a layout-by-layout basis, but rather a single set of table rules applies to every layout of a given or particular document type. This is not how most other technologies work. This approach allows the Paradatec solution to be relatively robust and impervious to layout changes over time.

Fuzzy Logic

Fuzzy Logic is crucial to the flexibility of the Paradatec solution. Fuzzy logic is heavily used in recognition of the keywords that we use to learn about most documents. For example, to locate the label "Claim Number" for a healthcare claim we may look for a series of synonyms for "Claim Number". On many recognition systems, poor image quality can result in a wrong result. For example, suppose the text analysis process located the text "C1aim Nunnber", i.e. the letter "l" is interpreted as the number "1" and the letter "m" is interpreted as the letter "n" twice. Paradatec uses "fuzzy matching" to determine if this text matches any of those sought. It will match this imperfect text with "Claim Number" but will give the result a slightly lower "fitness" than if it was sure that this is "Claim Number". Support for wildcards in our search strings makes things even easier. We can search for "Clai?" instead of "Claim" and the "?" matches any character.

Distinguishing Features of Paradatec Data Extraction

Paradatec data extraction has a number of distinguishing features when compared with other advanced recognition engines:

- Pre-Production Testing – is required by quality-conscious high-volume clients. Paradatec learning is performed off-line and is followed by rigorous regression testing and ONLY THEN are improved rules promoted to production. Other solutions that perform learning on-line in production have an inherent risk that a badly trained operator, or statistical outliers may allow the system to learn bad rules and infect the system with run time errors that are very difficult to find.
- Scalability – relating to layout variations. Because Paradatec treats each variation of a document (for example, "Mortgage Note" or "Paystub") as one document type, the software does not become slower in environments with thousands of variations of that document type. In systems that treat these variations as memorized layouts with layout-specific processing, the processing time is proportional to the number of variations (this is due to the fact that there is code that must test for each of the layouts prior to performing layout-specific processing).
- Simplicity – rules are configured by the person most familiar with the documents via a GUI. In other solutions, more complex operations require programming/scripting by an IT professional.
- Reliability – conducting full-page textual analysis on every page is the only reliable way to ensure not missing critical data. Some solutions use shortcuts to avoid the performance delays inherent in their full-page recognition engines.
- Reverse lookup – in environments where a database stores index information that may be expected to be printed on a document, Paradatec software performs a reverse-lookup to see if this information is, in fact present. This approach allows the software to realize automation levels that are unparalleled in the industry.
- Generic rules – are inherently impervious to layout changes that occur over time. The alternate approach is to remember layout variations and perform layout-specific processing. This will work fine until a layout changes.

Indexing/Verification

Paradatec's PROKEY verification engine is designed specifically for unstructured documents and allows:

- An operator to create/move document borders
- Presentation of multiple results (likely alternatives)
- Processing of compound data (tables, addresses, …)
- Table column swapping (with data preserved after the swap)
- Ability to perform complex verification in an application DLL (C#, C++ or VB available)
- Drop-down enumerated lists
- Type-ahead support
- Address keying (find vendor or lender via zip code)
- Keyboard controlled (limiting need for use of mouse operations)
- Document structure (automatically via PAGETYPE change or manually)

Methodology Comparison Matrix

	Judicious Learning	Visual Classification	Learning in Production
Requires document separator sheets between multi-page documents	No	Yes	Yes
Works with several capture platforms	Yes	No	No
Makes live changes to production operation	No	No	Yes
Needs scripting skills for complex configuration	No	Yes	Yes
Learning ability	Yes	Yes	Yes
Mailroom applicability	Yes	Yes	Yes
Mostly used in Accounts Payable	No	No	Yes
Predominantly Image-based or text-based	Text-based	Image then text-based	Text-based
Semi-structured or unstructured in approach	Unstructured	Semi-structured	Semi-structured
Sub-second full-page textual analysis	Yes	No	No