2011 International Conference on Document Analysis and Recognition

Automatic Content Extraction on Semi-Structured Documents

José Eduardo Bastos dos Santos
Perceptive Software
Shawnee, USA
joseeduardo.santos@perceptivesoftware.com

***Abstract*—Extracting specific content from certain types of documents can be a very challenging task, especially when developing a not so tailored solution and refraining from using explicit contextual information. In this paper, we address the problem of automatically extracting data from semi-structured documents through an unsupervised process based on an analysis of the document's own morphological composition. We also discuss how this approach can be applied to different types of documents, with special attention being paid to college transcripts. The success of our method is supported by extensive tests, from which we have drawn some authentic examples.**

***Keywords*-page decomposition; data extraction; document image understanding; automatic zoning; college transcripts, invoices, geometric and logical layout analysis.**

I. Introduction

Over the course of the last two decades, the world saw great progress in technologies pertaining to the automation of tasks related to document image processing. Usually guided by the problems caused by massive volumes of documents, applied science seems to have contemplated some areas more than others. For instance, postal automation [2], bank checks processing [5], and digital libraries [1] are a few well-founded examples among many others that saw a great deal of improvement in the automation of their processes.

In the field of Document Image Analysis, we can see that much has been done with page decomposition and particularly with separating text from non-text elements, either for documents or for video- and camera-based images. However, for *semi-structured documents*, where the structure is the same across different samples but the diversity of their content dictates their appearance, we are faced with an intriguing case of data capture. Despite its apparent uniformity, its segmentation imposes some unique demands due to the inconsistency and disposition of its content. Typical examples are line-item extraction from invoices or purchase orders, tax forms processing, as well as transfer credits and equivalencies processing from college transcripts. Needless to say that for each one of these data-intensive tasks, manual data entry is a time-consuming and error-prone activity that organizations should avoid for the sake of accuracy and productivity.

We often see systems performing data capture on semi-structured documents making use of contextual information. It is not uncommon to use templates to identify the document's structure, or OCR-based engines that search for key terms to locate specific content. And sometimes a combination of both approaches is used. However it is important to always consider the viability and the performance aspects of these solutions. The practical drawback of using templates arises from the variability of these documents since a new template must be generated for each new form. While this might be the case for most of the template based solutions, Medvet [10] shows a method to automatically update templates on an invoice processing system. The performance-related pitfall of using OCR-based engines comes from the dependency on quality images, which may not be the rule in a real-world environment dealing with a large-scale volume of documents. OCR-based solutions can also have a limited relevancy since they are language specific and dependent on the OCR's accuracy. In [8], Wnek uses a combination of template matching and OCR to extract data from documents such as insurance forms and invoices. The method learns the position and other geometric features from the data to be extracted throughout an inductive process that also involves the form's template. Takebe [9] presents a method to extract data from invoices and tax forms without analyzing the document's layout. The accuracy of their method varies depending on the task being performed: general data extraction or header only extraction.

The alternative we propose in this paper is a structural analysis that can accurately identify specific content from semi-structured documents by analyzing only its own composition. As we can see in the following sections, one of the challeges faced here is the fact that the main content on the document is usually blended with other non-relevant text lines and not isolated in individual blocks. Since the process is based on individual text lines and not blocks or groups of lines, isolating line items on invoices becomes a straightforward task. Although our method can be applied to different types of forms, we made our case using college transcripts, EOBs[1] and invoices. This is an unsupervised process whose main contributions are: 1) the complete absence of templates or models, 2) there is no need for a training phase, and 3) the capability of processing a broad

[1]Explanation Of Benefits: The document sent by the health insurance plan briefly detailing health services obtained and their reimbursements.

1520-5363/11 $26.00 © 2011 IEEE
DOI 10.1109/ICDAR.2011.249

IEEE computer society

range of documents.

Although invoices processing became a well-linked subject in the past years, very little can be found on college transcripts processing other than some commercial applications. However, the method we offer here comprises an original strategy with unparalleled matches on the literature. This method reaches a success rate of more than 95% on a database of 447 images of transcripts for the geometric layout analysis.

This paper presents the details of our work in the following manner: Section II defines the basics of data capture for college transcripts; Section III provides a description of how the document's structure can be used to identify specific content, along with substantial tests and examples; Sections IV and V conclude with discussion and identify some directions for future work.

II. College Transcripts Processing

Today in the United States there are approximately 4352 higher education institutions (colleges, universities and junior colleges), with only 70% of them having more than 500 students. Transfering credits from a previous university is the most common situation, leading to data extraction from academic transcripts. Although electronic college transcripts are available at some higher education institutions, regular paper transcripts still account for the vast majority of the transcripts processed every year by American universities. Althought there is no accurate information about the precise number of transcripts processed annually, it is known that only the states of California and Texas process together about one million transcripts annually. Thus, some institutions started experiencing bottlenecks in their admissions offices due to the high volume of transcripts, which in turn created a higher demand for efficiency, reduction of manual data entry, and easier record access.

A. Transcripts Processing Goal

Like most semi-structured documents, college transcripts are different from school to school. Even though the essential information is present in all transcripts, they vary in the amount of detail provided. From the approximately 250 different fields a transcript can accommodate, a much more restricted number is generally used for most colleges and universities. Commonly used transcript fields include the following data:

- **personal data:** name, date of birth, age, genre, address, student ID number or Social Security Number, etc.
- **previous degrees:** (when existent) school name, year, major, etc.
- **course data:** term, course codes, course descriptions, hours attempted, hours earned, possible points, points earned, grade, summary/totals, GPA, accumulated GPA, etc.

From the items above, course data represents the core information contained in a transcript and is the main target of data capture systems for transcripts.

B. Some Common Challenges

Automatically processing these type of documents can be a difficult task, especially for systems based on OCR and rules where the maxim *garbage in, garbage out* holds true. Template-based systems may undergo some difficulties covering the wide spectrum of possible layouts. Regardless of what the strategy, the differences in document layout and the disposition of data are two common problems in this domain. The orientation, number of document columns, content variation and background extraction are some expected difficulties.

III. Method Description

For the geometrical layout analysis process, we are interested to identify and correctly group similar text rows into meaningful arrangements so that these groups can posteriorly go through a logical classification. For college transcripts, the goal is to extract the course information from each semester block so that the school can gather all of the data about the courses a student attended, when the student atended the courses, and details related to the studend's grades, points, and hours. We look at each one of the semester blocks as an individual table or tabular array where the structural composition of each row is different. However, this difference is smaller among individual groups of rows. When we look at tables, we see how they communicate a document's content through the relationship of its rows and columns. And this mutual dependency is what better represents the concept we want to explore.

A. Structural Clustering

To get started, we pre-process the image by applying denoise, de-skew, and de-speckle processes. We also clean up the lines and borders, since the borders in some transcripts contain text that may interfere with the segmentation process. Once pre-processing is done, the image is analyzed so that its columns can be identified. Figure 1 shows a portion of a transcript and each one of the left-hand columns it contains. Because we do not need to read its content at this point, each word here is represented by a colored box where each color represents a different column. One can see, for example, that column 1 is present on rows $\{a, b, c, i, s\}$. Column 1 is therefore the catalyst element, bonding together all of these rows in a specific combination that can only be reproduced if all of these rows contain more than one column in common.

So, let $X(I) = \{x_i | i = 1 \dots n\}$ be the set of all columns x_i on image I and let Ω be the sample space of all possible row combinations according to the existing columns on each of these rows. Hence ω_1 is a sub-set of Ω containing rows

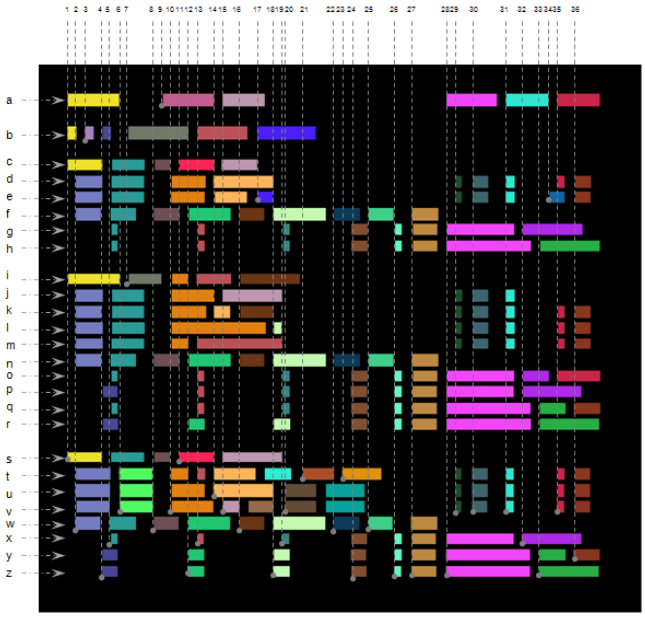

Figure 1. Each one of the left-aligned columns on a transcripts extract

$\{a, b, c, i, s\}$, the only rows with a block in column 1. In each ω_i only a smaller group of columns is responsible for representing the most cohesive group of rows according to their structural composition. However, in order to find these groups of very similar rows, we can clearly see that only parts of all columns efficiently help to identify them (figure 2). The excessive number of columns should be reduced to only the most representative columns in each group so that we can better address this classification problem. To identify these columns, we can analyze the relationship of all the columns in ω_i and look for an optimum set of columns according to the following rule:

$$g = \{y \in Y^{x_i} \mid p(y|\omega_i) > \tau\} \tag{1}$$

where Y^{x_i} represents the columns in ω_i and g represents a sub-set of columns whose probability, given the sub-set to which they pertain, ω_i, is above a suitable threshold dynamically calculated over ω_i. From figure 2 we see that for ω_{28} columns $\{24, 26, 27, 28\}$ are the most representative ones. Analyzing each column individually we try to build groups of rows that are structurally correlated. Each row might belong to different groups according to the columns belonging to it. These groups are defined based on the best, or more characteristic, set of columns they hold. That is to say, a synthetic binary row ρ is generated based on the columns from Y^{x_i} where each element from g is represented by 1 or 0 otherwise. A binary abstraction is produced for each row on ω_i and the dissimilarity between each of these binary rows and ρ is calculated. The final sub-group ω'_i will then include only the rows that are closer to ρ.

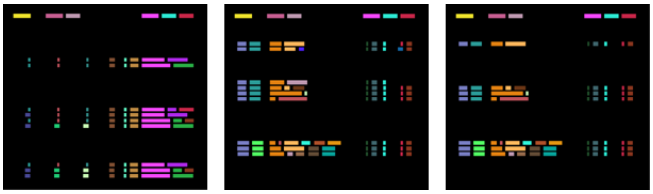

Figure 2. Three of the sub-spaces from Ω: ω_{28}, ω_{31} and ω_{35}

It is also important to say that even though figures 1 and 2 were generated based on left alignment only (for visual clarity), the core process may actually make use of both left and right alignments. For college transcripts, center alignment is not really useful, but right alignment is a common choice for numeric fields like hours attempted, hours earned, or points.

Once the first part of the process is complete, we are faced with a decision step. After defining more cohesive sub-groups of rows for each ω_i and considering the number of columns is usually larger than the number of rows, it is not unusual to see one row being part of more than one final sub-group ω'_i. For instance, if we look at row e we notice that it can be part of at least two distinct final sub-groups: ω'_2 and ω'_{14}. That means row e could be part of an arrangement with rows $\{d, e, j, k, l, m, t, u, v\}$ or within a smaller group with rows $\{d, e, k, t, u\}$. Having larger groups can help to speed up the logical classification of these groups, but we need to keep in mind that the compactness or homogeneity of each ω'_i is equally or more important than the final number of sub-groups. In order to decide the final group to which each row should belong, we could use a criterion based on the relative frequency of each class or even apply a cluster validity index [4] that can check for the compactness of each final sub-group along with the distances between them.

B. Matching Similar Patterns

As one might notice, the structural clustering process can be very specific, since the existence of an extra column can cause the process to classify similar rows into separated groups. At this point, we can look at these rows from a more distant perspective and use different criteria to see which rows can actually be re-assigned to one conjoint group. In doing so, we are trying to (a) speed-up the logical classification process by having the smallest possible number of classes and (b) lower the risk of having inconsistencies among groups that should receive the same logical classification.

Since structural clustering tries to group individual rows together, this part of the process goes a little further by analyzing the groups generated by structural clustering in an attempt to put some of them together. Two very simple ways of doing so run through the centroid of each ω'_i. We could either determine the correlation between two candidate centroids or calculate the distance between them. In this

case, two sub-groups ω'_p and ω'_q are re-grouped if the distance between their centroids are smaller than a specific value calculated over their own length.

It is important to emphasize that this second grouping is not as important as the "purity" of each individual group. Because the process is not intended to split groups of rows once they are formed, the quality and homogeneity of these groups is crucial in order to have a successful logical classification.

IV. Experiments and Results

The process was tested on different types of documents such as EOBs and invoices, but especially on college transcripts. For our database the most common layout for college transcripts is a two-column layout (65% of the total) with a white-space separator (64% against 36% with line separators). Transcripts with only one column follow right behind (34%), and documents with three or four columns are the most rare, representing less than 1% of the total. For our tests, a set of 447 images from 327 institutions was randomly picked from a larger collection of transcripts.

Because the layout of these documents can be so different for each school, it ends up playing an important role in the segmentation process. In our case, we first tried to detect the main area of the document, ignoring the top and bottom parts and breaking down each document column to be individually analyzed. The potential problem with this approach stems from documents with more than one document column where there is a small amount of data in the second or third columns. A low number of text rows means that there will be a low frequency of similar rows, which in turn tends to increase the number of groups due to the variability among those rows. This shows the importance of looking to each document column according to the document flow and not as separated entities [6]. This approach simplifies the structural clustering process in two ways: 1) it allows the data columns to be more populated, and 2) the discrimination between text row classes is more effective due to an increase in the frequency of the class, which also results in an increase in the inter-classes distance.

Table I summarizes the errors found during geometric layout analysis (structural clustering + matching similar patterns). Although we would like to have as few groups from the same class as possible, having multiple groups of rows for the same class (*course rows, summary rows* or *term rows*) is not necessarily a problem. However, we are concerned with the misclassification of text rows, which can essentially happen in three different ways. Let's call a *cloud row* any text row that is not a *term row, a course row* or a *summary row*. Therefore:

- **Type I errors** occur when a *cloud row* is misclassified as one of the main text rows.
- **Type II errors** occur when a main text row is classified as a *cloud row*.

Table I
Types of text rows on transcripts and the percentage of error for each one of them

	Summary	Course	Term
Occurrences	6	9	83
Type I	16.7%	–	93.97%
Type II	33.3%	11.1%	3.6%
Type III	50%	88.9%	2.4%
Relevant occurrences	6	9	5
Total errors	1.34%	2.01%	1.12%

- **Type III errors** occur when a main text row is classified as a different text row (such as a *course row* being classified as a *summary row)*.

As we can see, the problems with *term rows* are the most common among all the possible errors. This is due to the fact that these rows are usually very short, and consequently their structure can be easily mistaken for a different type of text row (usually some other short *cloud row*). However, almost 94% of the errors found with term rows are Type I errors. This means that typically these rows cannot cause major damage during logical classification, because they do not have a supporting structure of main text rows around them to build a semester block. Therefore, they are not relevant for this case. Consequently, the accuracy measured over our data set rises to 95.53%.

The tests performed over EOBs and invoices were different from the tests performed over transcripts, especially because of the smaller number of invoices and EOBs available. Nevertheless, the method responded well for both invoices and EOBs. Some modifications were made to the layout analysis due to there being less structural variation. Moreover, both types of documents can have one line item ending over the next line, which is an unusual scenario for transcripts.

In figure 3, we present a transcript image with the final grouping results. Not only were the main text rows correctly grouped, but all of the remaining cloud rows were disposed of in a coherent way. Although some sensitive data is masked for privacy reasons, the image clearly shows all of the important content and its final classification. We can see that all the course rows (purple) were successfully assigned to one single class, the same happening for summary rows (pink) and their headers (beige). Term rows were separated in three groups (red, blue and light blue) with no negative implications since they are not mixed with any other type of text row.

For invoices, we used a small group of 30 images for testing. Unlike college transcripts, invoices include much less variation in their layout because they have predominantly one document column. The orientation and the separation of each invoice column (solid lines or white space), however,

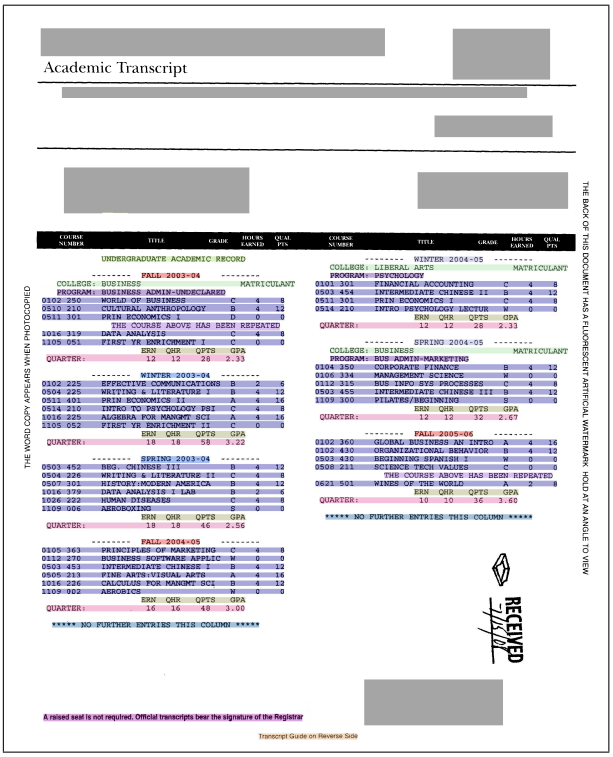

Figure 3. A real college transcript and the final result of our geometrical classification process

can still fluctuate. This is a real world set of images that include handwritten notes, stamps and bar codes. The set of images is also composed of invoices from different countries in different languages. Although invoice number and date are two commonly reviewed pieces of information, we focus on extracting line items and totals. This allows for a deeper analysis of each document and, consequently, a better financial control.

We verified only 2 mistakes among the 196 line items present in our database, which represents an accuracy of 98.97%. As for the totals, because these rows usually come with other information on the same text line (usually on their left side), they demand a closer analysis; the layout on the bottom portion of these documents can be more complicated than the rest of the page due to the presence of text lines of different heights, credit card logos, fine print return information, etc. For these reasons, the accuracy on totals reached only about 70%.

V. Conclusion

We have presented a method for extracting relevant content from semi-structured documents. In contrast to other methods that rely on templates, rules, or OCR-based engines, our method relies on documents having their content captured by an accurate analysis of their own morphological composition and the structural relationship between columns and text rows. Breaking it down into a two-step procedure, we first execute what we call a structural clustering, which is a comprehensive analysis of the document's columns, grouping similar text rows according to their homogeneity. Following that is a process that fine-tunes the results and merges together–not individual text rows–but groups of text rows. As a typical bottom-up procedure, we do not allow groups to be split, and so extra caution is taken when associating these text rows.

We tested the process over a broad set of different types of documents, and it shows very promising results at 95.53% accuracy for the used data set. Our next step is to feed a logical layout analysis with the results obtained by the geometric layout analysis in order to assign a logical meaning to each one of the classes found during geometrical analysis.

References

[1] L. Vincent, *Google Book Search: document Understanding on a Massive Scale*. Proc. of ICDAR2007, Curitiba, Brazil, 2007.

[2] P.W. Palumbo, S.N. Srihari, J. Soh, R. Sridhar, V. Demjanenko, *Postal address block location in real time*. IEEE Computer, Volume 25 Issue 7, July 1992.

[3] Hyung Il Koo and Nam Ik Cho, *State Estimation in a document Image and Its Application in Text Block Identification and Text Line Extraction* . Proc. of 11^{th} European Conference on Computer Vision, Crete, Greece 2010.

[4] Maria Halkidi, Yannis Batistakis, Michalis Vazirgiannis, *Cluster Validity Methods: Part I*. ACM SIGMOD Record, Volume 31 Issue 2, June 2002.

[5] N. Gorski, V. Anisimov, E. Augustin, S. Mysko, J.-C. Simon, *A new A2iA bankcheck recognition system*. Third European Workshop on Handwriting Analysis and Recognition, 1998.

[6] T. M. Breuel, *Two geometric algorithms for layout analysis*. In Workshop on document Analysis Systems, Springer-Verlag, 2002.

[7] Faisal Shafait, Joost van Beusekom, Daniel Keysers, Thomas M. Breuel, *Background Variability Modeling for Statistical Layout Analysis*. Proc. of 19^{th} International Conference on Pattern Recognition, Tampa, USA, 2008.

[8] Janusz Wnek *Automated Data Extraction from Structured documents*. Proc. of 2003 Symposium on document Image Understanding Technology, Greenbelt, USA, 2003.

[9] Hiroaki Takebe, Katsuhito Fujimoto, *Word Extraction Method by Generating Multiple Character Hypotheses*. Proc. of Eighth IAPR International Workshop on document Analysis Systems, Nara, Japan, 2008.

[10] Eric Medvet, Alberto Bartoli,Giorgio Davanzo, *A Probabilistic Approach to Printed Document Understanding*. International Journal on Document Analysis and Recognition