Analysis and Reporting of Results of Certification Exams for the Identification of Areas of Opportunity Part 2

Roger Ivan Osalde Cauich
Data Engineering
Universidad Politécnica de Yucatán
Km. 4.5. Carretera Mérida — Tetiz
Tablaje Catastral 4448. CP 97357
Ucú, Yucatán. México
Email: al1609097@upy.edu.mx

Lic. Alejandra Cabrera Casillas
Universidad Politécnica de Yucatán
Km. 4.5. Carretera Mérida — Tetiz
Tablaje Catastral 4448. CP 97357
Ucú, Yucatán. México
Email: Alejandra.cabrera@upy.edu.mx

Didier Omar Gamboa Angulo
Universidad Politécnica de Yucatán
Km. 4.5. Carretera Mérida — Tetiz
Tablaje Catastral 4448. CP 97357
Ucú, Yucatán. México
Email: didier.gamboa@upy.edu.mx

Abstract

The goal of this project was the automation of the information from the International Test of English Proficiency (iTEP) test, in order to identify the student needs at the Universidad Politécnica de Yucatán (UPY).

Identifying the sub-skills through their scores on the language knowledge and skills sections: Grammar, Listening and Reading so once the deficient sub-skills were detected, a strategy to improve the English program to reduce the failing scores can be implemented. The analysis was on ITEP test language skills and language knowledge sub-skills scores, the dataset is composed of the results of the students from 9 semester which were used for the validation tests and for the assist on cancel out the effects of unobserved factors. The use of frameworks in the Python programming language, allowed the reading and extraction of the results of the students, in a coherent and orderly way, which gives us the future opportunity to carry out various analyzes on the different categories that are evaluated within the exam ITEP, as they have been done in the past and in this way to be able to obtain the areas of use and risk within the UPY university community. The automation process seeks to reduce the work time for obtaining information, which can lengthen the time for the results of the analysis. Therefore, the teachers could be able to redesign the English program to support the student needs based on results.

Index Terms

Binary Mask: It is the set of data that, together with an operation, allows to selectively extract certain data stored in another set.

File extension: A file extension or file name extension is the ending of a file that helps identify the type of file in operating systems.

ITEP: International Test of English Proficiency or iTEP is a language assessment tool that measures the English skills of non-native English speakers.

Grayscale: Digital photography, computer-generated imagery, and colorimetry, a grayscale or image is one in which the value of each pixel is a single sample representing only an amount of light.

Analysis and Reporting of Results of Certification Exams for the Identification of Areas of Opportunity Part 2

I. Introduction

IN our days, technological advancement advances by leaps and bounds due to the needs of society that demand the improvement of processes and systems that provide help in daily tasks.

In all sectors it is necessary to have technological growth as well as a permanent improvement and optimization of all the processes or products that are created, among these sectors I can find the education sector. For technology to advance, and with it society, it is necessary for the people involved in the advances to carry out research and modifications in their processes to always have faster and higher quality results.

I speak of automation as any automatic intervention that helps industrial systems or processes to improve and optimize any process that is carried out. This may well be applied even to information collection programs to accelerate the processes of reading and extracting information, which is usually one of the longest processes to generate knowledge.

These processes allow us to answer relevant questions to describe and evaluate the results, which is the foundation of the hypothesis derived from our datasets. Nowadays, the handling of massive datasets has evolved so much that their analysis has become a primordial task; only in this way, the opportunities, and insights of descriptive analysis of the data can be obtained.

Teaching English to engineering students requires dynamic and planned methods for the development of tasks and challenges that contribute to the process of innovation to detect the areas of opportunity that instructors should adopt to improve the teaching of English according to the theoretical models of language.

II. Objectives

Generate automatic reports for the entrance of new data which detect the students' real needs to innovate the English Program in distinction to reduce the disapproved rate. These results are obtained from the ITEP exams, since is an international standardized test. It was possible to Generate an automated extraction process on the results of the students using Python frameworks to facilitate the reading processes in the ITEP results, which are generated in PDF format. However, the descriptive analysis of the university students could not be possible, due to the short time between the exams and the delivery of the final report of the stay.

III. State of the Art

In the markup there are several tools to process PDFs, I found two types of programs for manage PDF, the first type converts the PDF to other formats such as Microsoft Powers Points presentations and Microsoft Word documents, an example of this is PDFelement(Imagen 1) that in addition to converting the PDF to other formats, allows you to modify the PDF directly. This tool could be useful to open a PDF into a Word format and view the code of the PDF and extract information from there. This tools was not used in the project because for openning the ITEPS PDF in another format, PDFelement ask about the the encryption password, which I did not have.

Imagen 1

The second type of tools are those that extract data from the PDF when the data is in tables, an example is the ReportMiner, in our case only page 1 of the PDF is in tables, the following ones are not. I prefered use a Python library to extract the data in the tables, The library I chose instead of Report mines, it was Tabula-py.

imagen 2

IV. Methods and Tools

The development was the automatization of the information extraction from the dataset that are in PDF format. The dataset was collected through the ITEP exam's outcomes that students at Universidad Politecnica de Yucatan took to certificate their English level. Furthermore, a random sample was used to specific description of the opportunity areas

A. ITEP Academic

According to the ITEP International, [11]"the Listening, Reading, Writing, and Speaking sections of iTEP Academic align with the traditional categorization of language skills, and the Grammar section aligns with the notion of language knowledge."

B. Dataset

The scores were recorded by the English coordinator in an PDF file. The dataset structure involved both qualitative and quantitative variables to be analyzed, the columns are: overall scores and language sub-skills.

Variables:

1) Grammar sub-skills:

- Articles prepositions
- Conjunctions
- Expressing Quantity
- Parts of Speech
- Pronouns
- Sentence Structure
- Verb Forms

2) Listening sub-skills:

- Catching Details
- Connecting Content
- Determining the Context
- Main Idea
- Making Implications

3) Reading sub-skills:

- Detail
- Main Idea
- Sequencing
- Synthesis
- Vocabulary

4) Overall score (The values are: A2, B1, B2 and C1)

C. Tools

The following tools were used during the automation process, which significantly helped to speed up the extraction process given their qualities and the functions that they allow us to use during the process.

The set of libraries used during the extraction was:

- Tabula-py

Tabula-py is a simple Python wrapper of Tabula-py-java, which can read tables in a PDF. You can read tables from a PDF and convert them into a Pandas DataFrame. Tabula-py also enables you to convert a PDF file into a CSV, a TSV or a JSON file.

- CV2

Library to help the drawing process with OpenCV. Made to add labels to the images. Classification of images, etc.

- PIL

This library provides extensive file format support, an efficient internal representation, and fairly powerful image processing capabilities. The core image library is designed for fast access to data stored in a few basic pixel formats. It should provide a solid foundation for a general image processing tool.

- Pandas

Pandas is a Python package that provides fast, flexible, and expressive data structures designed with structured (Tabula-pyr, multidimensional, potentially heterogeneous) and time series data both easy and intuitive.

- Numpy

Is a library that supports creating large multidimensional matrices and vectors, along with a large collection of high-level mathematical functions to operate on them.

V. Development

A. Data Extraction

1) Rename PDF's: Our program has to process PDF with the same name scheme (examples: doc1, doc2, doc3, etc), it was noticed that the PDF comes with the name of the student that was presented that ITEP, or it comes with just the filename doc and the file extension.

To automate the people you order to download the PDF's, you don't have to put a specific name to each PDF. For this, the Rename function was created, which changes the name of all the PDF's that are in the folder where the notebook is, the name they put is to the examples above. The figure 3 show a example how the PDF should and should not be saved.

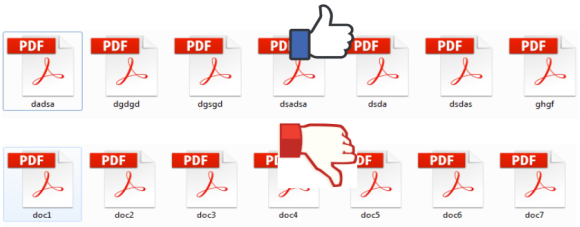

Figure 3

2) Extract data from tables-Pages 1: For the extraction of the Big skill and the level found on page 1, I will use the Tabula-py library, which I will only pass as parameters the name of the PDF I are analyzing and page 1 of this PDF. Tabula-py will return a data frame that contains the score and the level of each big skill, the scores are put into their respective list depending on the skill they represent, the student's level, I calculated it by taking an average of the levels of the 5 big skills.

All this process will be done iteratively for each PDF, for this process it was created the PDF function, which asks the user how many PDF's to analyze and enter that data as input. Then create the PDF function, which asks the user how many PDF's is going to analyze.

3) From PDF to JPEG: The first step was converted the pages of the PDF in images with the extension of jpeg[Figure 3]. This is because I used Open CV for analyzing where are the rectangles of the sub skills and what percentage represents.

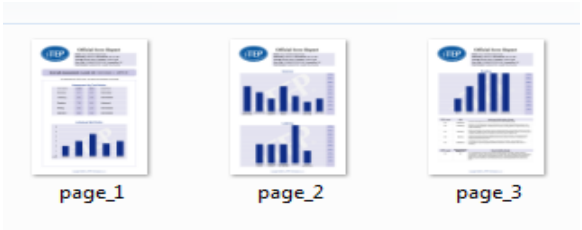

Figure 4

I read the images with Open CV, next I reduce the size of the image, this is because is more easy to Open CV analyze a small photo. I applied a scale of grays and the methods of canny, dilate and erode, next I draw in the image the contours of the figures that Open CV find. I print the real image with the contour that Open CV find[Figure 4], and the scale of grays imagen [Figure 5]

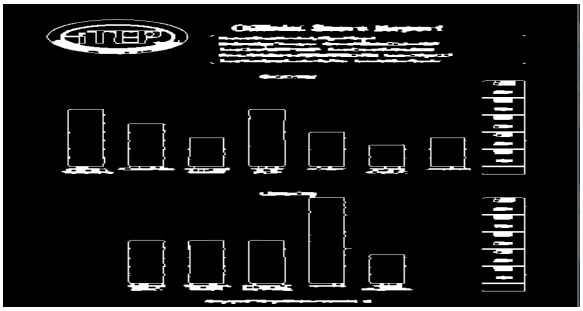

Figure 5

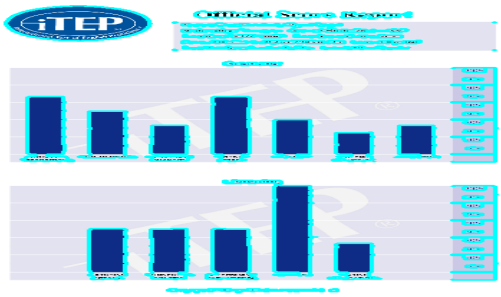

Figure 6

I seen that Open CV can detect the rectangles, so the next step was apply a mask blue, this is for only finding the figures that has the same blue color, we applied a canny effect and draw the contours, and print the image[Figure 6].

Figure 7

Now in the image only are the rectangles that I want, the next step is find what percentage the rectangles represent according to their areas, for this I try to find what what is the area of the rectangles that it represent 100 percentage. For this task I used the function cv.contourArea(). I found, the rectangles that represents 100 percentage of every sub skills they had different areas[Figure 7]:

Figure 8

The next step was found which sub skill represents each area, for this I extract the x position of every rectangle, I put the x position and the area in a list together. Once I found these x position, I write then in the image[Figure 8]

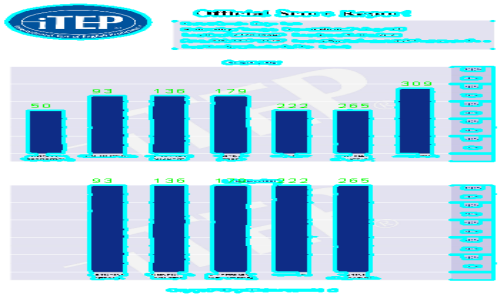

Figure 9

The extraction of area and the position in x of the blue rectangles was done for pages 2 and 3 of the PDF, for each one of these sheets a list was generated that contains in tuples a position in x and an area of each rectangle.

When the second page of the PDF is analyzed, a list of 12 tuples are obtained, every tuple contain a x position of one rectangle and its area. If the list contain less that 12 tuples that means that a skill is zero, you need to find the positions of the rectangles, what is missing is the missing skill.

23 lists were created which will save the data of each sub skill, big skill and the level. First, the data is entered for the sub skill of the second sheet, there are 12 sub skills, divided into two graphs, the graph at the top of the page has 7 sub skills and the graph below with 5 sub skills. Open CV reads from bottom to top, therefore, in the list of areas and you place the first 5 tuples will always be from the graph below, our program will take each tuple and check its positions in x, if the tuple is one of the first 5 means that the position in x is of a sub skill from the graph below, otherwise it will be from the graph above. After the program finds which sub-skill the area belongs to, it will pass the area to the function percentage, which will calculate what percentage that area represents. This percentage will be entered into the corresponding sub skill list.

The same will be done for page 3, but in this case the areas and positions found will only be one of the rectangles of a

single graph

VI. Results

In this section of results I will be able to observe the functions and parts of the program that were essential for the automation of the information extraction and a brief explanation about its operation, at the end of this section you will be able to observe a small example about the final result of the automation and as this ends in the creation of a file that will function as a database for future activities.

- Function PDF

It asks for the number of PDF's to analyze(N), with the help of a "for" loop it will apply the next function and task on n number of the PDFs in the folder. It take the N PDF's and convert all their page into images(3 images will be create, because the PDF is 3 pages long).

- Function x_area

Then introduce the images "page_2.jpeg" and "page_3.jpeg" in the function "x_area" which will come back in a list the position of every rectangle in the image and the area of these rectangles. These lists are introduced in the functions of calculate_porcentage1 and calculate_porcentage2, which they will calculate the percentage that the area of the rectangles represents and find which sub skill the rectangles represents. Next it will call the function "check_list" which will check when a skill does not any value and introduce a zero to its respectable list when that is true. Next the function read the first page of the PDF with Tabula-py, with this function I will extract the big skills ("Level","Grammar","Listening","Reading","Writing", "Speaking"), for finishing some list are cleaned for the next PDF, and I deleted the image that function create.

- Function calculate_porcentage1

It receive the list that the function x_area created, the function will take the x values for find with sub skills represent any area, next when it find the sub-skill it will calculate the percentage that the area represents, the percentage will be introduced in the list of sub skill that it present. and will append a respectably number to the list "list_habilides_score", this is for check when a skill is zero.

- Function Percentage

Function Percentage: the area that it presents the 100 percentage of every sub skill has different measure for all the sub skills. This function receives the area of a sub skills, and a number that represents the sub skills, with the number I found the measure that represent 100 percentage of every sub skills, and with this I calculated the percentage that the area represents.

- Function check_list

This function check if the number that represents every sub skill appear in the list "list_habilides_score", if any number does not appear, it means that the value of that sub skill is zero, if this happens the function will put a zero in the respectable list of that sub skill.

- Function tablaa

This function receive the page 1 of the PDF that I read with Tabula-py, I extract the score of all big skill and the function puts then in a list, next with the score the function calculates the level, that is an average of big skills.

The final results of the application of the functions is the generation of an Excel file in which the data of the PDf's contained in the folders next to the program notebook are stored. As can be seen in the image, the information that is stored in the Excel file are the percentages of the sub skills' grades as well as the general percentages of the Grammar, Listening, Reading, Writing and Speaking skills, together with the final percentages of the ITEP The final level of the students is attached, which ranges from 1 to 6 according to their domain of the English language. It was done in this way to maintain the confidentiality of the students' results as well as the personal information of those who took the test, since only their final results are necessary to be able to carry out a later analysis.

International Test English Proficiency					
Level	Grammar	Listening	Reading	Writing	Speaking
3.5	52	42	100	70	70
3.6	68	71	70	70	62

TABLE I
ITEP RESULTS

The Source Code is available in the GitHub Repository of the project[12]. The code has comments enough to understand the steps and the processes made it in the project of amortization extraction.

VII. Conclusion

Automating information extraction from PDF files generated by the ITEP exam is a step forward in improving data and information handling processes. The objective of the project was achieved thanks to the use of the different tools to which I have access right now, the use of these tools allowed generating a program capable of solving the need that arose when having to record this information in an Excel file which functions as a database for the analysis of opportunities and risk areas in English language learning.

It is worth mentioning that the tools used in this program are Open Source, so it was possible to avoid generating an additional cost to the university in the extraction of this information. The program has a simple structure and is sufficiently documented so that it can be reused and even improved in the future for subsequent projects that seek to carry out a deeper automation in the university's systems.

Although the information extraction process is complete, I want to continue with the project to identify areas of risk and opportunity, carrying out the necessary statistical analyzes to continue supporting the improvement of the student community in the command of the English language.

The only recommendation that I will mention is that the one or those who are going to use the program have the basic knowledge of Python and how to make this program work, in short, that they know how to run the program in order to obtain the data.

Finally, the members of the project would like to work more in depth on the university's automation systems, from the lowest levels, to be able to create an embedded system between departments and registers, in which the level of errors is seen significantly decreased and thus be able to avoid possible problems such as those faced by some students during their tests.

REFERENCES

[1] OpenCv modules,Opencv Open Source Computer Vision, Online:https://docs.opencv.org/master/

[2] P. Joshi,OpenCV with Python By Example,Packt Publishing,2015

[3] Azzi R., Despres S., Diallo G. (2020) KEFT: Knowledge Extraction and Graph Building from Statistical Data Tables. In: Hernes M., Wojtkiewicz K., Szczerbicki E. (eds) Advances in Computational Collective Intelligence. ICCCI 2020. Communications in Computer and Information Science, vol 1287. Springer, Cham

[4] Librery os Miscellaneous operating system interfaces, python documentation, https://docs.python.org/3/library/os.html

[5] Lopez Luis Yu, Jingyi Arighi, Cecilia Huang, Hongzhan Shatkay, Hagit Wu, Cathy. (2011). An Automatic System for Extracting Figures and Captions in Biomedical PDF Documents.

[6] Déjean, Hervé Meunier, Jean-Luc. (2009). On tables of contents and how to recognize them.

[7] Christopher Clark and Santosh Divvala.(2016).Looking Beyond Text: Extracting Figures, Tables and Captions from Computer Science Papers.The Allen Institute for Artificial Intelligence

[8] Noah Siegel, Nicholas Lourie, Russell Power, Waleed Ammar.(2018). Extracting Scientific Figures with Distantly Supervised Neural Networks.Allen Institute for Artificial Intelligence

[9] Christopher Clark and Santosh Divvala.(2017).Mining Figures from Research Papers.University of Washington.

[10] Piotr Adam Praczyk, Javier Nogueras-Iso and Salvatore Mele.(2013). Automatic Extraction of Figures from Scientific Publications in High-Energy Physics.Universidad de Zaragoza, Spain.

[11] ITEP International, http://www.itepexam.com/wp-content/uploads/2016/10/iTEP Academic Reliability Validity(06OCT16).pdf

[12] Source Code from the data extraction. Available in: https://github.com/RogerOsalde/Intership_2_Winter