

Towards Automated Assessment of Students' Preliminary Thesis Submissions

Matúš Sulír and Jaroslav Porubán
Department of Computers and Informatics
Faculty of Electrical Engineering and Informatics
Technical University of Košice
Letná 9, 042 00 Košice, Slovakia
Email: {matus.sulir, jaroslav.poruban}@tuke.sk

Abstract—While a bachelor's or master's thesis is an important document, students often leave their elaboration to the last minute. We established a system of continuous thesis submissions: Students upload their preliminary theses to a learning management system at least six times an academic year. The submissions are assessed for formal criteria like a minimum page count, citation style or inclusion of a particular chapter. In the past, whole evaluation was performed manually, which was a laborious, time-consuming activity. We managed to automate some parts of this process. In this paper, we describe an approach used for automation and lessons learned during application of the system.

I. INTRODUCTION

A bachelor's or master's thesis is the most important work of a student of the given degree. However, students tend to postpone a large part of the work on their thesis to the last moment. This is natural since the Parkinson's first law states the work expands to fill the available time [1].

To at least partially suppress this behavior, we established a system of multiple work-in-progress thesis submissions during an academic year.

In order to be effective, each submission should be evaluated as well. For fair judgment, a set of criteria was formulated. The description of this system, the criteria, advantages and disadvantages are comprised in the first part of the article.

In the past, all aspects of theses were evaluated manually, which was an error-prone and time-consuming task. For example, we needed to count filled pages in documents manually. We have managed to automate some parts of this process. The description of this automation and future directions are described in the second part of the article.

II. CONTINUOUS SUBMISSION

A. Courses

The approach was applied on the students of the Informatics study program, which is attended by more than a hundred students per a year of study.

1) *Bachelor's Degree*: For the bachelor's degree, there are two courses which have thesis elaboration as the main goal: Bachelor Project in the winter term (WT) and Bachelor Thesis in the summer term (ST) of the third year. There are 6 submission rounds total – three in the winter term and three in the summer one.

2) *Master's Degree*: The master's program has three thesis-relevant courses: Master's Project 1, 2 and Master's Thesis. While the first one uses continuous submission too, it is only evaluated as submitted / not submitted. Therefore, we will focus on the second and third course. The structure is same as for the bachelor's degree – 6 submission rounds total, 3 in the winter and 3 in the summer semester.

B. Submission System

There is one course created in the Moodle¹ LMS (learning management system) for the bachelor's thesis and one for the master's thesis. Students can submit current versions of their theses anytime, ideally each week. The mentioned six submissions are mandatory and limited by strict deadlines. Students upload their thesis in a form of a PDF (Portable Document Format) file, plus other attachments if necessary.

C. Criteria

After each deadline, the submission round is evaluated. Because of a large number of students, only formal criteria can be assessed globally. The content aspect of a thesis is assessed by a supervisor as usual.

A summary of the criteria is in Table I.

1) *Page Count*: The number of pages includes only manually written sections – for example, Introduction is included, but Table of Contents is not. Furthermore, if a page is not fully filled, only the filled proportion is counted. For instance, a page half-filled with text and images is counted as 0.5 pages.

2) *References*: The next criterion is the number of bibliographic references. The presence of many references can be an indicator of thorough literature review. However, their quality should be also taken into account.

The criterion "References good" is fulfilled when:

- the bibliographic references are mostly research papers,
- there are not too much direct Wikipedia citations
- and the references are in a good citation style according to the standards [2].

The references should be "well-cited", too. Principally, this means:

¹<http://moodle.org>

TABLE I
CRITERIA FOR WORK-IN-PROGRESS THESIS SUBMISSIONS

Bachelor's thesis					
Submission round	Deadline	Min. pages (total)	Min. references (total)	Content	Other
1	6th week WT	8	10	analytical part	in template, good references, well-cited
2	12th week WT	12	15	motivation, goals, analytical part	
3	WT exam period	18	-	methods, conceptual design	presentation
4	6th week ST	24	-	detailed design, implementation	
5	10th week ST	30	-	evaluation, contribution	
6	ST exam period	30	-	introduction, conclusion	attachments as in a final version
Master's thesis					
Submission round	Deadline	Min. pages (total)	Min. references (total)	Content	Other
1	6th week WT	10	15	analytical part	in template, good references, well-cited
2	12th week WT	20	20	motivation, goals, analytical part	
3	WT exam period	30	-	methods, conceptual design	presentation
4	5th week ST	40	-	detailed design, implementation	
5	7th week ST	50	-	evaluation, contribution	
6	9th week ST	50	-	introduction, conclusion	attachments as in a final version

- There are no dangling references – items listed in the Bibliography section, but not used in the thesis text at all.
- The literature is cited directly where it is used, i.e., not at the end of a paragraph.

3) *Content*: For each submission round, there is a list of thesis parts that should be elaborated. The most notable is the analytical part, which is planned for 12 weeks. It is followed by a research method, design, implementation description and a very important part – evaluation.

4) *Other*: In the first submission round, we check whether the theses are in the mandatory school-supplied document template (the criterion “in template”).

Some submissions require attachments. For each semester, a student should prepare a presentation which will be then presented in front of a supervisor. For the last submission, the student should also upload all attachments like the program developed with its source code, user and system guide, etc.

D. Advantages

The main purpose of preliminary submissions was to encourage students to work on their theses continuously. This goal was successful because the majority of students upload their thesis in time. Nevertheless, the Parkinson's rule holds: although submissions are open practically all time, students rarely upload their works sooner than the last week before the deadline.

If some formal criteria are not met, students are given the chance to correct their mistakes before the final evaluation. This, in connection with the first advantage, should lead to higher quality theses.

Since the submissions are assessed, they can be easily compared. It is possible to create a list of all students, compare them or even assign parts of grades according to the results.

The long period left for the analytical part gives a student enough time for “market research” – to find out users' needs, explore existing solutions and find their shortcomings. In the past, students often neglected this important part of each thesis and skipped directly to implementation of a system – which could already exist, more usable, stable and feature-rich.

E. Disadvantages

As any teaching or assessment method, our continuous thesis submission approach is not without drawbacks.

First of all, some students may get the feeling that having a thesis which fulfills all formal criteria is a sole sufficient condition to defend it.

The most controversial criterion is the minimum page limit. Students are then often tempted to write “everything about nothing”. This is in contrast with the fact that brevity is one of the most appreciated properties of scientific and technical writing [3].

III. MANUAL ASSESSMENT

Now we will describe the original process of fully manual assessment.

A. Evaluation Tables

For each submission round, there exist a specific spreadsheet. Rows in such table represent individual students and their theses. In the columns, there are criteria. An example of an evaluation table for the first bachelors' submission is in Table II.

First of all, the thesis must be submitted before the deadline. Otherwise, it will not be assessed at all. The more submissions a student will miss, the worse his/her best grade can be. Missing one submission per term means the student cannot receive a grade better than C for the given course, for two

TABLE II
AN EXCERPT OF A SAMPLE EVALUATION TABLE

Student	Thesis name	Submitted	In template	8 pages	10 references	Good references	Well-cited	Analytical part	Comment
A. Apple	Barification	x	x	x	x		x	x	online sources only
B. Carrot	Frobnication	x	x			x		x	wrong citation style
C. Jam	Foobaz								
W. Juice	Frob Bar	x	x	x	x	x	x	x	
...									

missed submissions it is E. Ignoring all preliminary thesis submissions means FX (fail).

The next columns represent criteria mentioned in Table I.

Note that every criterion is assessed in a boolean manner: true or false. If a criterion is fulfilled, there is a mark “x” in the spreadsheet. When it is unmet, there is no mark in the given cell.

The last column can contain a free-form comment explaining how specifically were the criteria unmet, what is a possible remedy and other notes.

In addition, the spreadsheet is conditionally formatted using colors. Each row with all criteria fulfilled is marked with green background. Yellow and pink color means some criteria are not met. Red color means the student missed or ignored the submission. This way, we can gain a visual insight into success or failure to meet formal criteria.

B. Assessment Process

Before any automation took place, we proceeded as follows: The spreadsheet was divided into multiple smaller files, each one containing a portion of students. These smaller files were assigned to the personnel (PhD students and teaching assistants). These persons downloaded the theses from Moodle, checked and marked each cell in the table manually. Filled partial tables were then reassembled into one spreadsheet.

The number of theses is large: in order of hundreds for each submission round. Assessing all theses completely manually wasted valuable time of highly qualified personnel. Therefore, we decided to at least partially automate the process.

IV. ASSESSMENT AUTOMATION

We designed the automation to be compatible with manual checking. This way, the process can be gradually more automated.

The automation is performed by a Ruby script. It works in a batch command-line mode, i.e., it is run with appropriate arguments, processes an input and writes an output.

A. Input

The first input is an evaluation spreadsheet file. The file must already contain student names/theses as rows and criteria for the given submission round as columns. The true/false cells should be empty, as they will be overwritten.

The next argument is a ZIP file which contains all theses as PDFs. This file can be easily obtained by using

the Moodle “Download all assignments as a zip” functionality. The files inside it are named in the form `Name Surname_filename.pdf`, so they are easily identifiable.

The last, optional argument is a ZIP file containing attachments like source code files or presentations. This is used only for some submission rounds. In Moodle, students upload attachments in a separate assignment (upload field).

B. Output

The output of the program is a partially filled spreadsheet with some criteria checked. The original spreadsheet file is overwritten with the resulting one.

Criteria supported for automation are:

- Is the thesis submitted?
- Does it have the specified minimum number of pages?
- Is the thesis in the mandatory template?
- Did the student upload any attachments?

The rest of the criteria must be checked manually.

C. Process

An overview of the automation algorithm follows.

1) *Header Recognition*: As we can see, it is not necessary to enter a submission round number. The script reads the first row in the spreadsheet – the header which contains criteria names. Thus, it is simple to recognize which criteria are evaluated in this submission round and which column represents a specific criterion. When the criterion is parametrized, the parameter is also recognized. For example, if the column is named “12 pages”, the program parses this string and finds out this is the criterion “minimum page count” with the parameter equal to 12.

2) *Name–Thesis Matching*: The script reads the cell values in the column “Student”, finds the corresponding entry in the ZIP file and extracts the thesis PDF into a temporary files directory.

When duplicate student names are found, the assessment of all duplicates is skipped and left for manual checking.

3) *Criteria Checking*: Each thesis is matched against the given criteria. This process is described in detail in section V.

4) *Result Writing*: The results are written back to the spreadsheet. The possible cell values are:

- “x” – for a fulfilled criterion,
- “” (an empty string) – for an unmet criterion,
- “?” – for duplicate student names and criteria whose fulfillment could not be reliably determined.

V. AUTOMATED CRITERIA CHECKING

Now we will describe methods used to check the supported criteria.

A. Submitted

If there exists a PDF file matching the student's name, the thesis was uploaded and the criterion is met, otherwise not.

B. Template

In the first submission round, it is checked whether the uploaded theses are written in one of the templates supplied by the university library. We manually extracted a few mandatory word phrases present in all templates. The program determines whether all these phrases are present in a thesis. If yes, the criterion is fulfilled, otherwise not. Note that we have two sets of phrases: one for Slovak templates and one for English ones.

If the uploaded thesis is not written in a template, we do not proceed to the criterion "page count", as the algorithm is template-dependent. We count the page count manually instead.

C. Page Count

As we already mentioned in section II-C1, just counting the total number of pages in the whole PDF file is insufficient. Furthermore, given the limitations of a PDF file format and a variety of subtle differences in the students' theses produced from different programs, this is the most complicated criterion to check.

1) *Main Content Range*: We count pages only from the Introduction chapter (including) up to the Bibliography/References chapter (excluding). The PDF format has a built-in support for chapters. Unfortunately, documents uploaded by our students, generated by a variety of PDF-producing programs, often do not have these metadata available.

It is therefore necessary to identify chapters by its top-level headings. A PDF file format defines just appearance, not semantics of text and graphics, unless it is a tagged PDF file [4], which is not our case. Hence, we must extract headings from the whole text using some criteria.

First, the PDF file is converted to an XML document using the `pdftohtml` utility². This XML document contains elements representing pages, fonts and text blocks. Each text block is characterized by its position, size and font used.

To identify top-level chapter headings, we currently use an XPath rule selecting text boxes with a height larger than 20 pt. To find the specific heading, e.g., Introduction, we have to match against a regular expression, since the students' theses can include a variety of headings with subtle differences, for example:

- Introduction
- 1 Introduction
- 1.Introduction
- 1. Introduction

²<http://pdftohtml.sourceforge.net>

Even worse, some students do not begin their thesis with a chapter called Introduction (although it is highly recommended), but with a chapter of a different name. For this reason, we also consider a chapter beginning with number "1" an introduction, if the Introduction chapter itself is not present.

We include both Slovak and English variants of chapters when performing the matching.

To improve robustness, we perform an accent-insensitive string comparison: diacritical marks are removed from their base letters both in the thesis text/XML and in the patterns. Occasionally, accents are separated from their base letters, e.g., *a'* instead of *á*. These standalone accents are removed, too.

Once we have identified the Introduction and Bibliography chapters, we obtain the page numbers which they were found on. In cases when the identification can not be performed, the thesis must be checked manually.

2) *Sample Pages Exclusion*: The thesis templates supplied by the university library include sample pages containing examples of headings, texts, images, tables, etc., illustrating their use. Students very often upload their theses with these sample pages included. We must differentiate between the sample content and the student's own content. Otherwise, it could largely influence the assessed page count: Some templates contain more than 10 sample pages.

A list of small excerpts from the sample texts was created. We randomly selected two or three sequences of approximately 15 words from each sample page. We were careful to choose sequences which would occur very unlikely in students' own works.

Each page in the tested thesis which contains one of these sample strings is then considered a sample page. We decided to exclude all recognized sample pages from the assessed page count. This approach has a disadvantage: If a sample text is included on the same page with the student's own content, it may be unfairly excluded from the counted pages. However, this situation occurs very rarely.

At the end of this stage, the script possesses a set of page numbers. They represent pages (we will call them *content pages*) of a thesis which will be further processed.

3) *Filled Page Portion*: In cases where each chapter is on a separate page, there can be many pages which are almost blank. Furthermore, Word often places images to suboptimal positions, leaving too much blank space. Therefore, for each page, we must determine the filled percentage.

First, all content pages of a PDF file are converted into bitmap images for simpler processing.

Let us look at Fig. 1 which depicts the page layout of a thesis written in the university template. A particularly large portion is covered by a header and footer. For our calculation to be more precise, we cut out the header and footer beforehand. Fortunately, they are clearly separated from the rest of a page by solid lines, which are graphically recognized by our script. If the script can not find a line, it cuts off 8% of a page from each side.

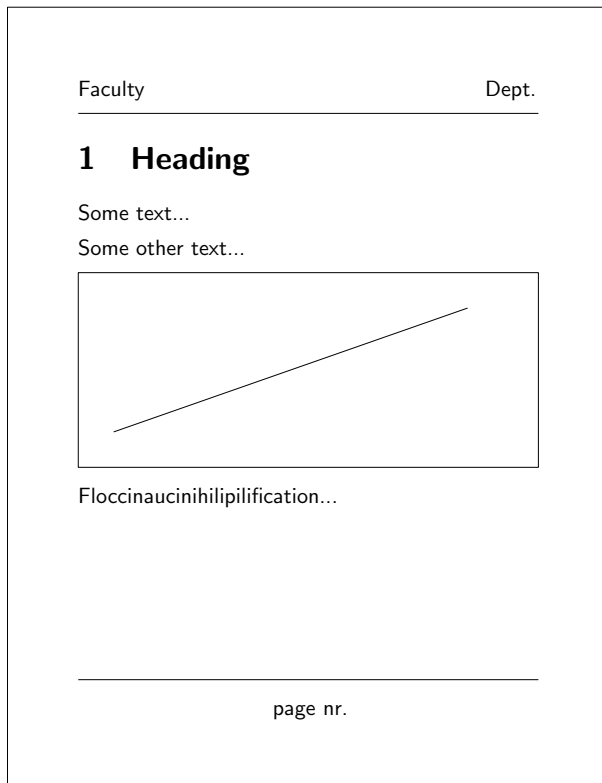


Fig. 1. The thesis page layout.

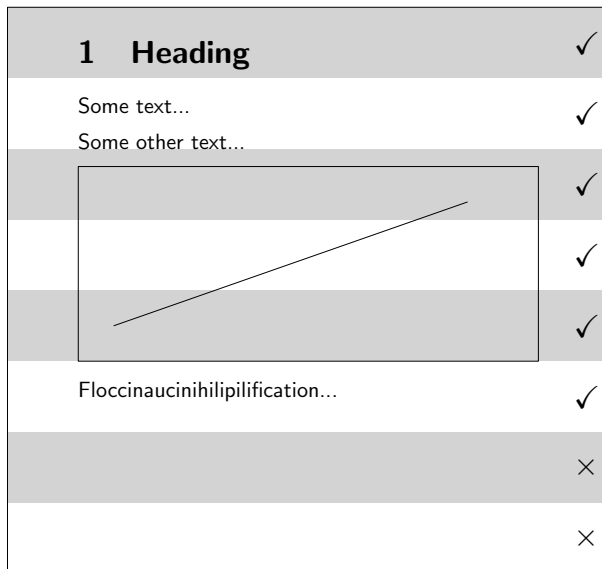


Fig. 2. The filled page portion calculation.

The script takes the resulting bitmap and divides it into 8 equally high stripes. See Fig. 2 for illustration.

For each of these stripes, it determines whether it contains something. We consider an image non-empty if the sum of inverted pixel values (because of the white color to become 0) reaches a given threshold value.

Now we have a number of non-empty stripes for a given

page. To get the page filled ratio, we divide the number of non-empty stripes by the total number of stripes (8). The example in Fig. 2 contains 6 filled stripes, so the filled portion is 0.75.

4) *Total Page Count*: Once the script determines the filled portion of each page, it sums all these values. The resulting value is the total page count.

We assess the minimum page count criterion with a negative tolerance of 2%. For instance, if the criterion is to have at least 10 pages, we accept 9.8 pages or more.

D. Attachments

Some submission rounds include attachments upload. Currently, the script only checks whether there is any file present in the attachments ZIP for the given student. We do not check a file extension or even its contents.

VI. LESSONS LEARNED

Apart from a method of semi-automatic thesis assessment, the contribution of this article includes some important lessons learned during the testing of our program.

A. Text Instructions Are Insufficient

Students can write their thesis in either $\text{\LaTeX}$ or Word / LibreOffice. Therefore, their source files can be in the DOC, DOCX, ODT or (possibly zipped) TEX format. We require the students to upload the generated PDF file into Moodle. This simplifies further analysis: an analyzer for one file format is much simpler than for multiple formats. Furthermore, the DOC and DOCX files are often mangled when viewed in different versions of programs than they were created.

Directly on the Moodle submission page, there is a highly visible note that only PDF files are supported. Regretfully, many students ignored the PDF-only requirement and uploaded their source files instead. Although the the number of such students gradually lowered (23, 18 and 8 during three consecutive submissions), it did not reach zero.

The best solution would be to restrict the uploaded file type in Moodle for the given assignments. However, this is not possible in its current version. Initially we accepted also non-PDF files as “submitted” with a friendly notice in the evaluation table that the PDF file should be uploaded. Currently we consider a non-PDF file “not submitted”.

B. Templates Do Not Mean Consistency

Although students are required to use the supplied templates to write their theses, the files they upload to Moodle often formally differ from each other.

The differences range from subtle mistakes like renaming a standardized chapter like Introduction, through variance of heading formats, to layouts totally scattered and not similar to the template at all.

It is hard to tell what are the reasons of these inconsistencies. We did not investigate which programs and their versions the students use to write theses and to produce PDFs.

The $\text{\LaTeX}$ template seem to be the most reliable one – it discourages its users to change e.g., the formatting of section

headings. Encouraging students to use the $\text{\LaTeX}$ typographical system, for instance by showing them its advantages and by rewarding them with points, seems to be a good start.

C. Parkinson's Law Is Powerful

While we motivate students not to delay their duties to the latest time possible, the Parkinson's first law still holds.

According to an anecdotal evidence, 4 days before the final deadline of the first submission, just around 1% of bachelor's degree and 7% of master's degree students already uploaded their thesis to Moodle.

Moreover, some students leave uploading to the last minutes, or even seconds before the deadline.

VII. RELATED WORK

A. Written Works Assessment

Burstein et al. [5] present a web-based system for automated essay evaluation. It is able to detect grammatical and stylistic errors and score the students' written works according to these formal criteria.

Our criteria are different than the ones of Burstein et al. Presently, we do not include grammatical and stylistic errors as a criterion in any of the submission rounds.

Furthermore, the input to our system is a document containing not only text, but also tables, images, etc. Our goal is mainly to count pages in a document in its visual form. The mentioned essay evaluation system considers only plain text.

B. Programming Assignment Evaluation

In [6], a system of automated testing of students' programming assignments is described. They use continuous monitoring of students' progress to achieve higher motivation and spot the errors sooner than during the final evaluation.

Similar systems exist for distributed programming [7] and assembly languages [8].

C. Other

To count the filled page percentage, it may seem suitable to use the method used to count the cartridge yield for printers [9], i.e., to determine the total portion of a page filled by some color. However, a whole page containing just text could take up less area of color than a particularly small solid-filled image.

VIII. CONCLUSION AND FUTURE WORK

A. Conclusion

We described:

- an approach of continuous students' bachelor's and master's theses submission
- and a method of their semi-automated assessment and evaluation.

The advantages of continuous students' theses submission are:

- To eliminate the Parkinson's law and motivate students to work continuously.

- To spot formal errors in the theses sooner than during the final evaluation which allows them to correct their works.

On the other hand, students may be tempted to write too lengthy theses because of the minimum page limit.

By automating the most laborious parts of thesis assessment:

- we significantly lowered the time used to assess thesis,
- saved the precious time of teachers and researchers
- and made evaluation more fair.

Our script is not bullet-proof and is prone to errors, especially when dealing with documents diverging from the templates. In addition, even if all criteria were assessed automatically, we would have to check theses manually because of a possibility of intentional rules circumvention by students.

B. Future Work

The most important future work is to automate assessment of more criteria. Namely, these criteria could be checked automatically:

- the minimum number of references,
- correctness of the citation style ("well-cited").

In this article, we presented just an overview of the continuous submission system and a brief description of an automation script. Because the system (particularly the automation script) is relatively new, we were not able to collect enough data to perform rigorous analysis of students' performance. A comparison of results of manual and semi-automated checking of the same set of theses is also an interesting idea for future work.

ACKNOWLEDGMENT

This work was supported by project KEGA No. 019TUKE-4/2014 Integration of the Basic Theories of Software Engineering into Courses for Informatics Master Study Programmes at Technical Universities – Proposal and Implementation.

REFERENCES

- [1] L. A. Brannon, P. J. Hershberger, and T. C. Brock, "Timeless demonstrations of Parkinson's first law," *Psychonomic Bulletin & Review*, vol. 6, no. 1, pp. 148–156, 1999.
- [2] *Guidelines for bibliographic references and citations to information resources*, ISO Std. 690, 2010.
- [3] L. D. Yore, B. M. Hand, and M. K. Florence, "Scientists' views of science, models of writing, and science writing practices," *Journal of Research in Science Teaching*, vol. 41, no. 4, pp. 338–369, 2004.
- [4] M. R. B. Hardy and D. F. Brailsford, "Mapping and displaying structural transformations between XML and PDF," in *Proceedings of the 2002 ACM Symposium on Document Engineering*, ser. DocEng '02. New York, NY, USA: ACM, 2002, pp. 95–102.
- [5] J. Burstein, M. Chodorow, and C. Leacock, "Automated essay evaluation: The Criterion online writing service," *AI Mag.*, vol. 25, no. 3, pp. 27–36, Sep. 2004.
- [6] M. Novák and M. Biñas, "Automated testing of case studies in programming courses," in *Emerging eLearning Technologies and Applications (ICETA)*, 2011 9th International Conference on, Oct 2011, pp. 157–162.
- [7] M. Paralič and J. Martončík, "Support for management of programming assignments - automated grading," in *Emerging eLearning Technologies Applications (ICETA)*, 2012 IEEE 10th International Conference on, Nov 2012, pp. 285–289.
- [8] S. Šimoňák, "Assignment workflow support system," *Acta Electrotechnica et Informatica*, vol. 12, no. 1, pp. 25–29, 2012.
- [9] *Method for the determination of toner cartridge yield for monochromatic electrophotographic printers and multi-function devices that contain printer components*, ISO/IEC Std. 19752, 2004.