Texas A&M University - Commerce

Department of Computer Science

Advancements in Forest Fire Prediction: Integrating Artificial Intelligence and Statistical Inference

Mounika Malka

Supervisor: Derek Harter, Ph.D.

A report submitted in partial fulfilment of the requirements of
Texas A&M University - Commerce for the degree of
Master of Science in *Computer Science*

February 12, 2024

Declaration

I, Mounika Malka, of the Department of Computer Science, Texas A&M University - Commerce, confirm that this is my own work and figures, tables, equations, code snippets, artworks, and illustrations in this report are original and have not been taken from any other person's work, except where the works of others have been explicitly acknowledged, quoted, and referenced. I understand that if failing to do so will be considered a case of plagiarism. Plagiarism is a form of academic misconduct and will be penalised accordingly.

I give consent to a copy of my report being shared with future students as an exemplar.

I give consent for my work to be made available more widely to members of TAMUC and public with interest in teaching, learning and research.

<div align="right">

Mounika Malka
February 12, 2024

</div>

Abstract

Forest fires pose significant threats to ecosystems, human lives, and infrastructure. Predicting forest fire occurrence is crucial for effective resource allocation, mitigation, and recovery efforts. This paper explores recent advancements in forest fire prediction methodologies, mainly focusing on integrating artificial intelligence (AI) and statistical inference techniques. We discuss the implications of reduced parameter sets in AI-based models for efficient prediction systems, especially pertinent to developing countries. Moreover, we delve into the statistical properties of random forest models, shedding light on their error distributions and potential for statistical inference. Through a comprehensive literature review and comparative analysis, we aim to provide insights into cutting-edge approaches for forest fire prediction, paving the way for more accurate and reliable prediction systems.

Keywords: Forest fire occurrence prediction, Support vector machines, Artificial neural networks, Feature Reduction, Weather data

Acknowledgements

An acknowledgements section is optional. You may like to acknowledge the support and help of your supervisor(s), friends, or any other person(s), department(s), institute(s), etc. If you have been provided specific facility from department/school acknowledged so.

Contents

List of Figures

List of Tables

List of Abbreviations

SMPCS   School of Mathematical, Physical and Computational Sciences

Chapter 1

Introduction

1.1 Background

Forest fires represent a significant threat to ecosystems, human lives, and infrastructure worldwide. These catastrophic events result in immediate devastation and long-term environmental and socioeconomic impacts [4]. Forest fires' increasing frequency and severity, particularly in regions with hot, dry climates, have underscored the urgency of developing effective prediction and management strategies [5]. Understanding the factors contributing to forest fire occurrence and progression is essential for mitigating risks and minimizing damages.

1.2 Problem Statement

The challenge of accurately predicting forest fires lies in the complex interactions between various environmental factors, including weather conditions, vegetation types, and human activities. Conventional prediction systems often rely on extensive monitoring features and weather prediction mechanisms, which can be costly and impractical, especially for developing countries. Furthermore, weather prediction inaccuracies can lead to fire risk assessment errors [2]. Therefore, there is a pressing need for cost-effective and efficient forest fire prediction methods that can reliably estimate fire occurrence and progression.

1.3 Aims and Objectives

The primary aim of this study is to investigate and evaluate machine learning techniques for forest fire prediction, focusing on enhancing prediction accuracy and efficiency. The specific objectives of the project are as follows:

- Review and analyze existing forest fire prediction methodologies, including traditional systems and artificial intelligence-based approaches [4].

- To assess the performance of the developed models using real-world forest fire data and evaluate their effectiveness in predicting fire occurrence and progression [5].

1.4 Solution Approach

This project adopts a comprehensive approach to address the challenges associated with forest fire prediction. The methodology involves:

- Reviewing relevant literature on forest fire prediction methods and machine learning techniques.

- Implementing and fine-tuning machine learning models based on the identified methodologies.

- Collecting and prepossessing real-world forest fire data for model training and evaluation.

- Analysing the performance of the developed models and comparing them with existing prediction systems.

1.5 Summary of contribution and achievements

This paper contributes to the field of forest fire prediction by exploring various artificial intelligence-based methods and their applications. Specifically, it examines the genetic programming in predicting forest fire occurrences and estimating the extent of burned areas. By analyzing existing literature and conducting experiments, this paper provides insights into the strengths and limitations of different prediction models, offering valuable guidance for future research and practical implementation.

1.6 Organization of the report

The report starts with an Introduction where we discuss the topic's background, explain the problem we're trying to solve, outline our goals, and describe how we plan to solve the problem. Then, we have a Literature Review section where we review what others have written about our topic and explain how we've cited their work. The Methodology section explains the methods we used in our research. The Results section tells you what we learned from our study. Next, in the Discussion and Analysis section, we carefully consider our results, why they're essential, and mention any limitations we encountered. The Conclusions section summarizes the main things we discovered and suggests ideas for future research. Finally, in the Appendices, we include extra stuff like tables or more details for people who want to know more.

Chapter 2

Literature Review

2.1 Introduction

Predicting forest fire sizes is essential for implementing effective mitigation strategies and minimizing their destructive impact. In recent years, data mining techniques and meteorological data analysis have emerged as promising approaches for forest fire prediction. This literature survey examines notable studies in this domain, which propose various data-driven and climate-based models for predicting forest fire sizes. Through a comprehensive analysis, this survey aims to provide insights into the methodologies employed, their strengths and limitations, and opportunities for further research to enhance forest fire prediction accuracy and facilitate proactive management and mitigation efforts.

2.2 Example of in-text citation of references in LaTeX

A study found that 21.9Different scales are used for each of the FWI elements, high values suggest more severe burning conditions (Taylor and Alexander 2006)

2.3 Example of "risk" of unintentional plagiarism

Unintentional plagiarism arises when writers neglect to properly acknowledge borrowed information, often due to oversight. One common scenario involves the omission of citations for widely recognized facts or common knowledge within a specific field or context. For example, failing to attribute the fact that "water boils at 100 degrees Celsius at sea level" can inadvertently lead to plagiarism, even though it is widely acknowledged. This oversight, particularly in academic or formal writing, underscores the importance of diligently crediting all sources to maintain integrity and avoid unintentional plagiarism.

2.4 Critique of the review

The review provides an extensive analysis of methodologies employed in forest fire prediction, spanning data mining techniques, meteorological variables, and machine learning algorithms. [6]

notably focused on investigating various data mining techniques, particularly Support Vector Machines (SVM), to forecast forest fire sizes. While their study highlighted the effectiveness of SVM, a deeper critique is warranted regarding the challenges associated with implementing these techniques, including data availability, model complexity, and computational requirements . [7] hybrid model, integrates clustering and classification techniques, presents promising outcomes in forest fire prediction. Their approach, while innovative, lacks a comparative analysis with existing methodologies to fully elucidate its strengths and weaknesses. Furthermore, the review overlooks external factors like climate change and land-use patterns, which could significantly impact predictive accuracy Additionally [8] explores on the application of Random Forests, emphasizing ensemble methods' potential in capturing intricate relationships between meteorological variables and fire occurrence. While their study offers valuable insights, a deeper examination of the interpretability and robustness of Random Forest models is needed. Furthermore, discussing the scalability of these algorithms and their suitability for real-time prediction in large-scale forest areas would provide practical implications for forest fire management. A research on the influence of climate change on forest fire regimes underscores the importance of incorporating climate projections into predictive models[9] . Their emphasis on considering long-term trends and variability in climate parameters is noteworthy. However, the review could elaborate on the specific methodologies proposed for integrating climate data into predictive models and discuss challenges related to climate model uncertainty and downscaling techniques. Additionally, exploring the implications of changing fire weather patterns on forest fire behavior and the effectiveness of current mitigation strategies would enrich the discussion and provide valuable insights for future research.

2.5 Summary

The exploration of methodologies for forest fire prediction reveals promising avenues through data mining techniques, meteorological variables, and machine learning algorithms. While studies showcase the effectiveness of Support Vector Machines (SVM), hybrid models, and Random Forests, there remains a need for a deeper critique of their limitations and challenges, including data availability, model complexity, and scalability. Moreover, the significance of considering external factors like climate change and land-use patterns is evident, urging the integration of climate projections into predictive models. Addressing these aspects will be pivotal in advancing forest fire management and mitigation strategies.

Chapter 3

Methodology

We mentioned in Chapter 1 that a project report's structure could follow a particular paradigm. Hence, the organization of a report (effectively the Table of Content of a report) can vary depending on the type of project you are doing. Check which of the given examples suit your project. Alternatively, follow your supervisor's advice.

3.1 Examples of the sections of a methodology chapter

A general report structure is summarised (suggested) in Table 3.1. Table 3.1 describes that, in general, a typical report structure has three main parts: (1) front matter, (2) main text, and (3) end matter. The structure of the front matter and end matter will remain the same for all the undergraduate final year project report. However, the main text varies as per the project's needs.

3.1.1 Example of a software/Web development main text structure

Notice that the "methodology" Chapter of Software/Web development in Table 3.2 takes a standard software engineering paradigm (approach). Alternatively, these suggested sections can be the chapters of their own. Also, notice that "Chapter 5" in Table 3.2 is "Testing and Validation" which is different from the general report template mentioned in Table 3.1. Check with your supervisor if in doubt.

3.1.2 Example of an algorithm analysis main text structure

Some project might involve the implementation of a state-of-the-art algorithm and its performance analysis and comparison with other algorithms. In that case, the suggestion in Table 3.3 may suit you the best.

3.1.3 Example of an application type main text structure

If you are applying some algorithms/tools/technologies on some problems/datasets/etc., you may use the methodology section prescribed in Table 3.4.

Table 3.1: Undergraduate report template structure

Frontmatter		Title Page
		Abstract
		Acknowledgements
		Table of Contents
		List of Figures
		List of Tables
		List of Abbreviations
Main text	Chapter 1	Introduction
	Chapter 2	Literature Review
	Chapter 3	Methodology
	Chapter 4	Results
	Chapter 5	Discussion and Analysis
	Chapter 6	Conclusions and Future Work
	Chapter 7	Refection
End matter		References
		Appendices (Optional)
		Index (Optional)

Table 3.2: Example of a software engineering-type report structure

Chapter 1	Introduction	
Chapter 2	Literature Review	
Chapter 3	Methodology	
		Requirements specifications
		Analysis
		Design
		Implementations
Chapter 4	Testing and Validation	
Chapter 5	Results and Discussion	
Chapter 6	Conclusions and Future Work	
Chapter 7	Reflection	

3.1.4 Example of a science lab-type main text structure

If you are doing a science lab experiment type of project, you may use the methodology section suggested in Table 3.5. In this kind of project, you may refer to the "Methodology" section as "Materials and Methods."

Table 3.3: Example of an algorithm analysis type report structure

Chapter 1	Introduction	
Chapter 2	Literature Review	
Chapter 3	Methodology	
		Algorithms descriptions
		Implementations
		Experiments design
Chapter 4	Results	
Chapter 5	Discussion and Analysis	
Chapter 6	Conclusion and Future Work	
Chapter 7	Reflection	

Table 3.4: Example of an application type report structure

Chapter 1	Introduction	
Chapter 2	Literature Review	
Chapter 3	Methodology	
		Problems (tasks) descriptions
		Algorithms/tools/technologies/etc. descriptions
		Implementations
		Experiments design and setup
Chapter 4	Results	
Chapter 5	Discussion and Analysis	
Chapter 6	Conclusion and Future Work	
Chapter 7	Reflection	

Table 3.5: Example of a science lab experiment-type report structure

Chapter 1	Introduction	
Chapter 2	Literature Review	
Chapter 3	Materials and Methods	
		Problems (tasks) description
		Materials
		Procedures
		Implementations
		Experiment set-up
Chapter 4	Results	
Chapter 5	Discussion and Analysis	
Chapter 6	Conclusion and Future Work	
Chapter 7	Reflection	

3.2 Example of an Equation in LaTeX

Eq. 3.1 [note that this is an example of an equation's in-text citation] is an example of an equation in LaTeX. In Eq. (3.1), s is the mean of elements $x_i \in \mathbf{x}$:

$$s = \frac{1}{N} \sum_{i=1}^{N} x_i.$$

(3.1)

Have you noticed that all the variables of the equation are defined using the **in-text** maths command $.$, and Eq. (3.1) is treated as a part of the sentence with proper punctuation? Always treat an equation or expression as a part of the sentence.

3.3 Example of a Figure in LaTeX

Figure 3.1 is an example of a figure in LaTeX. For more details, check the link:
wikibooks.org/wiki/LaTeX/Floats,_Figures_and_Captions.
Keep your artwork (graphics, figures, illustrations) clean and readable. At least 300dpi is a good resolution of a PNG format artwork. However, an SVG format artwork saved as a PDF will produce the best quality graphics. There are numerous tools out there that can produce vector graphics and let you save that as an SVG file and/or as a PDF file. One example of such a tool is the "Flow algorithm software". Here is the link for that: flowgorithm.org.
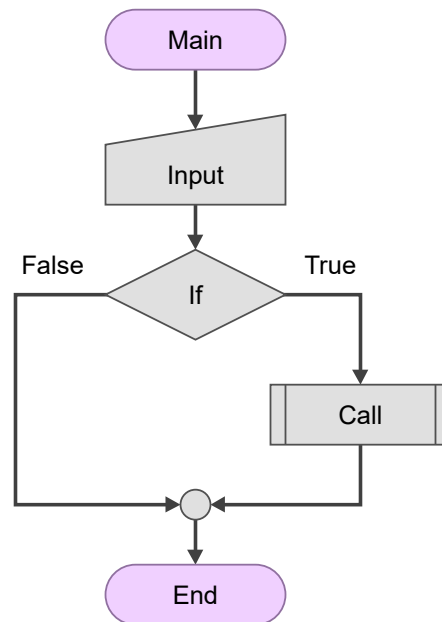

Figure 3.1: Example figure in LaTeX.

3.4 Example of an algorithm in LaTeX

Algorithm 1 is a good example of an algorithm in LaTeX.

Algorithm 1 Example caption: sum of all even numbers

Input: $\mathbf{x} = x_1, x_2, \ldots, x_N$
Output: $EvenSum$ (Sum of even numbers in $\mathbf{x}$)

1: **function** EVENSUMMATION($\mathbf{x}$)
2:    $EvenSum \leftarrow 0$
3:    $N \leftarrow length(\mathbf{x})$
4:    **for** $i \leftarrow 1$ to N **do**
5:        **if** $x_i \mod 2 == 0$ **then**   ▷ check if a number is even?
6:            $EvenSum \leftarrow EvenSum + x_i$
7:        **end if**
8:    **end for**
9:    **return** $EvenSum$
10: **end function**

3.5 Example of code snippet in LaTeX

Code Listing 3.1 is a good example of including a code snippet in a report. While using code snippets, take care of the following:

- do not paste your entire code (implementation) or everything you have coded. Add code snippets only.

- The algorithm shown in Algorithm 1 is usually preferred over code snippets in a technical/-scientific report.

- Make sure the entire code snippet or algorithm stays on a single page and does not overflow to another page(s).

Here are three examples of code snippets for three different languages (Python, Java, and CPP) illustrated in Listings 3.1, 3.2, and 3.3 respectively.

```python
1  import numpy as np
2
3  x   = [0, 1, 2, 3, 4, 5] # assign values to an array
4  evenSum = evenSummation(x) # call a function
5
6  def evenSummation(x):
7      evenSum = 0
8      n = len(x)
9      for i in range(n):
10         if np.mod(x[i],2) == 0: # check if a number is even?
11             evenSum = evenSum + x[i]
12     return evenSum
```

Listing 3.1: Code snippet in LaTeX and this is a Python code example

Here we used the "\clearpage" command and forced-out the second listing example onto the next page.

```
1  public class EvenSum{
2      public static int evenSummation(int[] x){
3          int evenSum = 0;
4          int n = x.length;
5          for(int i = 0; i < n; i++){
6              if(x[i]%2 == 0){ // check if a number is even?
7                  evenSum = evenSum + x[i];
8              }
9          }
10         return evenSum;
11     }
12     public static void main(String[] args){
13         int[] x  = {0, 1, 2, 3, 4, 5}; // assign values to an array
14         int evenSum = evenSummation(x);
15         System.out.println(evenSum);
16     }
17 }
```

Listing 3.2: Code snippet in LATEX and this is a Java code example

```
1  int evenSummation(int x[]){
2      int evenSum = 0;
3      int n = sizeof(x);
4      for(int i = 0; i < n; i++){
5          if(x[i]%2 == 0){ // check if a number is even?
6              evenSum = evenSum + x[i];
7          }
8      }
9      return evenSum;
10 }
11
12 int main(){
13     int x[]  = {0, 1, 2, 3, 4, 5}; // assign values to an array
14     int evenSum = evenSummation(x);
15     cout<<evenSum;
16     return 0;
17 }
```

Listing 3.3: Code snippet in LATEX and this is a C/C++ code example

3.6   Example of in-text citation style

3.6.1   Example of the equations and illustrations placement and reference in the text

Make sure whenever you refer to the equations, tables, figures, algorithms, and listings for the first time, they also appear (placed) somewhere on the same page or in the following page(s). Always make sure to refer to the equations, tables and figures used in the report. Do not leave them without an **in-text citation**. You can refer to equations, tables and figures more them once.

3.6.2   Example of the equations and illustrations style

Write **Eq.** with an uppercase "Eq" for an equation before using an equation number with (\eqref{.}). Use "Table" to refer to a table, "Figure" to refer to a figure, "Algorithm" to

refer to an algorithm and "Listing" to refer to listings (code snippets). Note that, we do not use the articles "a," "an," and "the" before the words Eq., Figure, Table, and Listing, but you may use an article for referring the words figure, table, etc. in general.

For example, the sentence "A report structure is shown in **the** Table 3.1" should be written as "A report structure is shown **in** Table 3.1."

3.7 Summary

Write a summary of this chapter.

Note: In the case of **software engineering** project a Chapter "**Testing and Validation**" should precede the "Results" chapter. See Section 3.1.1 for report organization of such project.

Chapter 4

Results

The results chapter tells a reader about your findings based on the methodology you have used to solve the investigated problem. For example:

- If your project aims to develop a software/web application, the results may be the developed software/system/performance of the system, etc., obtained using a relevant methodological approach in software engineering.

- If your project aims to implement an algorithm for its analysis, the results may be the performance of the algorithm obtained using a relevant experiment design.

- If your project aims to solve some problems/research questions over a collected dataset, the results may be the findings obtained using the applied tools/algorithms/etc.

Arrange your results and findings in a logical sequence.

4.1   A section

...

4.2 Example of a Table in LATEX

Table 4.1 is an example of a table created using the package LATEX "booktabs." do check the link: wikibooks.org/wiki/LaTeX/Tables for more details. A table should be clean and readable. Unnecessary horizontal lines and vertical lines in tables make them unreadable and messy. The example in Table 4.1 uses a minimum number of liens (only necessary ones). Make sure that the top rule and bottom rule (top and bottom horizontal lines) of a table are present.

Table 4.1: Example of a table in LATEX

Bike		
Type	Color	Price (£)
Electric	black	700
Hybrid	blue	500
Road	blue	300
Mountain	red	300
Folding	black	500

4.3 Example of captions style

- The **caption of a Figure (artwork) goes below** the artwork (Figure/Graphics/illustration). See example artwork in Figure 3.1.

- The **caption of a Table goes above** the table. See the example in Table 4.1.

- The **caption of an Algorithm goes above** the algorithm. See the example in Algorithm 1.

- The **caption of a Listing goes below** the Listing (Code snippet). See example listing in Listing 3.1.

4.4 Summary

Write a summary of this chapter.

Chapter 5

Discussion and Analysis

Depending on the type of project you are doing, this chapter can be merged with "Results" Chapter as " Results and Discussion" as suggested by your supervisor.

In the case of software development and the standalone applications, describe the significance of the obtained results/performance of the system.

5.1   A section

Discussion and analysis chapter evaluates and analyses the results. It interprets the obtained results.

5.2   Significance of the findings

In this chapter, you should also try to discuss the significance of the results and key findings, in order to enhance the reader's understanding of the investigated problem

5.3   Limitations

Discuss the key limitations and potential implications or improvements of the findings.

5.4   Summary

Write a summary of this chapter.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

Typically a conclusions chapter first summarizes the investigated problem and its aims and objectives. It summaries the critical/significant/major findings/results about the aims and objectives that have been obtained by applying the key methods/implementations/experiment set-ups. A conclusions chapter draws a picture/outline of your project's central and the most signification contributions and achievements.

A good conclusions summary could be approximately 300–500 words long, but this is just a recommendation.

A conclusions chapter followed by an abstract is the last things you write in your project report.

6.2 Future work

This section should refer to Chapter 4 where the author has reflected their criticality about their own solution. The future work is then sensibly proposed in this section.

Guidance on writing future work: While working on a project, you gain experience and learn the potential of your project and its future works. Discuss the future work of the project in technical terms. This has to be based on what has not been yet achieved in comparison to what you had initially planned and what you have learned from the project. Describe to a reader what future work(s) can be started from the things you have completed. This includes identifying what has not been achieved and what could be achieved.

A good future work summary could be approximately 300–500 words long, but this is just a recommendation.

Chapter 7

Reflection

Write a short paragraph on the substantial learning experience. This can include your decision-making approach in problem-solving.

Some hints: You obviously learned how to use different programming languages, write reports in LATEXand use other technical tools. In this section, we are more interested in what you thought about the experience. Take some time to think and reflect on your individual project as an experience, rather than just a list of technical skills and knowledge. You may describe things you have learned from the research approach and strategy, the process of identifying and solving a problem, the process research inquiry, and the understanding of the impact of the project on your learning experience and future work.

Also think in terms of:

- what knowledge and skills you have developed

- what challenges you faced, but was not able to overcome

- what you could do this project differently if the same or similar problem would come

- rationalize the divisions from your initial planed aims and objectives.

A good reflective summary could be approximately 300–500 words long, but this is just a recommendation.

Note: The next chapter is "**References**," which will be automatically generated if you are using BibTeX referencing method. This template uses BibTeX referencing. Also, note that there is difference between "References" and "Bibliography." The list of "References" strictly only contain the list of articles, paper, and content you have cited (i.e., refereed) in the report. Whereas Bibliography is a list that contains the list of articles, paper, and content you have cited in the report plus the list of articles, paper, and content you have read in order to gain knowledge from. We recommend to use only the list of "References."

References

Bedia, J., Herrera, S., Gutiérrez, J. M., Benali, A., Brands, S., Mota, B. and Moreno, J. M. (2015), 'Global patterns in the sensitivity of burned area to fire-weather: Implications for climate change', *Agricultural and Forest Meteorology* **214**, 369–379.

Carvalho, A., Flannigan, M. D., Logan, K., Miranda, A. I. and Borrego, C. (2008), 'Fire activity in portugal and its relationship to weather and the canadian fire weather index system', *International Journal of Wildland Fire* **17**(3), 328–338.

Castelli, M., Vanneschi, L. and Popovič, A. (2015), 'Predicting burned areas of forest fires: an artificial intelligence approach', *Fire ecology* **11**(1), 106–118.

Cortez, P. and Morais, A. d. J. R. (2007), 'A data mining approach to predict forest fires using meteorological data'.

Sakr, G. E., Elhajj, I. H. and Mitri, G. (2011), 'Efficient forest fire occurrence prediction for developing countries using two weather parameters', *Engineering Applications of Artificial Intelligence* **24**(5), 888–894.
URL: *https://www.sciencedirect.com/science/article/pii/S0952197611000418*

Sakr, G. E., Elhajj, I. H., Mitri, G. and Wejinya, U. C. (2010), Artificial intelligence for forest fire prediction, *in* '2010 IEEE/ASME international conference on advanced intelligent mechatronics', IEEE, pp. 1311–1316.

Shabbar, A., Skinner, W. and Flannigan, M. D. (2011), 'Prediction of seasonal forest fire severity in canada from large-scale climate patterns', *Journal of Applied Meteorology and Climatology* **50**(4), 785–799.

Shidik, G. F. and Mustofa, K. (2014), Predicting size of forest fire using hybrid model, *in* 'Information and Communication Technology: Second IFIP TC5/8 International Conference, ICT-EurAsia 2014, Bali, Indonesia, April 14-17, 2014. Proceedings 2', Springer, pp. 316–327.

Wager, S. (2014), 'Asymptotic theory for random forests', *arXiv: Statistics Theory* .
URL: *https://api.semanticscholar.org/CorpusID:41610136*
Wager (2014)
Sakr et al. (2011)
Sakr et al. (2010)
Shabbar et al. (2011)
Castelli et al. (2015)
Cortez and Morais (2007)

Shidik and Mustofa (2014)
Bedia et al. (2015)
Carvalho et al. (2008)

Appendix A

An Appendix Chapter (Optional)

Some lengthy tables, codes, raw data, length proofs, etc. which are **very important but not essential part** of the project report goes into an Appendix. An appendix is something a reader would consult if he/she needs extra information and a more comprehensive understating of the report. Also, note that you should use one appendix for one idea.

An appendix is optional. If you feel you do not need to include an appendix in your report, avoid including it. Sometime including irrelevant and unnecessary materials in the Appendices may unreasonably increase the total number of pages in your report and distract the reader.

Appendix B

An Appendix Chapter (Optional)

...