Texas A&M University - Commerce

Department of Computer Science

Unmasking Deception through Advanced NLP Analysis

Najeebuddin Mohammed

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 2, 2024

Declaration

I, Najeebuddin Mohammed, 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.

Najeebuddin Mohammed
February 2, 2024

Abstract

In the pursuit of countering the pervasive issue of misinformation in online articles, this project delves into the realm of Natural Language Processing (NLP). The aim is to bolster the precision of detecting fake news through a fusion of advanced linguistic analysis, sentiment modeling, and machine learning algorithms. The project's inception addresses the overarching context of misinformation and outlines a clear mission to mitigate its influence. Methodologies deployed encompass a comprehensive exploration of NLP techniques, feature selection strategies, and model optimization intricacies. The ensuing analysis scrutinizes the efficacy of NLP in identifying distinctive linguistic patterns associated with fake news sources. A pivotal aspect of the study involves investigating the nuanced impact of feature selection and model optimization on the overall performance of the fake news detection system. Conclusions drawn underscore the significance of the achieved results and their far-reaching implications for the ongoing battle against misinformation. The abstract concludes by underscoring the unique contribution of this project in advancing the domain of fake news detection through the lens of NLP methodologies.

Keywords: Natural Language Processing, fake news detection, linguistic analysis, machine learning, misinformation.

Report's total word count: we expect a maximum of 10,000 words (excluding reference and appendices) and about 10 pages. [A good project report can also be written in approximately 5,000 words.]

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

The pervasive dissemination of misinformation in online articles presents a critical challenge in today's information landscape. This project delves into the intersection of Natural Language Processing (NLP) and machine learning to address this issue. Motivated by the increasing impact of fake news on public perception and decision-making, the project seeks to contribute to the development of robust mechanisms for detecting and mitigating the spread of deceptive content.

1.2 Problem statement

Misinformation, especially in the form of fake news, poses a significant threat to information integrity. This section details the nuances of the investigated problem, emphasizing the need for accurate and efficient fake news detection mechanisms. Rather than a generalized exploration, the focus here is on a detailed articulation of the challenges associated with identifying deceptive content in online articles.

1.3 Aims and objectives

Aims: The central aim of this research is to enhance the precision of fake news detection within a corpus of news articles. Through the integration of NLP techniques and a diverse array of machine learning classifiers, the research aims to develop a specialized framework that considers the unique linguistic patterns and contextual characteristics of news articles.

Objectives:

1. Evaluate the effectiveness of NLP techniques in discerning distinctive linguistic patterns associated with fake news sources within the defined news article corpus.

2. Investigate the impact of sentiment modeling on understanding the emotional context of news articles and its potential contribution to fake news detection.

3. Assess the relevance and efficacy of machine learning classifiers in the context of news articles, considering factors such as feature selection and model optimization.

1.4 Solution approach

Briefly describe the solution approach and the methodology applied in solving the set aims and objectives.

Depending on the project, you may like to alter the "heading" of this section. Check with you supervisor. Also, check what subsection or any other section that can be added in or removed from this template.

1.4.1 A subsection 1

You may or may not need subsections here. Depending on your project's needs, add two or more subsection(s). A section takes at least two subsections.

1.4.2 A subsection 2

Depending on your project's needs, add more section(s) and subsection(s).

A subsection 1 of a subsection

The command \subsubsection{} creates a paragraph heading in LaTeX.

A subsection 2 of a subsection

Write your text here...

1.5 Summary of contributions and achievements

Describe clearly what you have done/created/achieved and what the major results and their implications are.

1.6 Organization of the report

Describe the outline of the rest of the report here. Let the reader know what to expect ahead in the report. Describe how you have organized your report.

Example: how to refer a chapter, section, subsection. This report is organised into seven chapters. Chapter 2 details the literature review of this project. In Section 3...

Note: Take care of the word like "Chapter," "Section," "Figure" etc. before the LaTeXcommand \ref{}. Otherwise, a sentence will be confusing. For example, In 2 literature review is described. In this sentence, the word "Chapter" is missing. Therefore, a reader would not know whether 2 is for a Chapter or a Section or a Figure.

Chapter 2

Literature Review

The realm of social media, encompassing forums, social networking, microblogging, social bookmarking, and wikis (*Using Social Media*, n.d.)(Gil, 2019), significantly influences the dynamics of information dissemination. However, the unintentional factors contributing to the rise of fake news, as evidenced by incidents like the Nepal Earthquake case (Tandoc Jr et al., 2017)(Radianti et al., 2016), underline the intricacies of navigating the digital information landscape. In 2020, the global health sector encountered a substantial surge in fake news, prompting the World Health Organization (WHO) to declare an 'infodemic' during the COVID-19 outbreak. This infodemic involved a flood of both authentic and false information, including a noteworthy volume of misinformation.

In response to the challenges of identifying and combatting fake news, several research initiatives have proposed innovative solutions. Sahoo and Gupta (2021) introduced an automatic fake news identification technique tailored for the Chrome environment, providing a means to detect fake news on Facebook. This approach leverages various features associated with a Facebook account, coupled with news content features, utilizing deep learning to analyze account characteristics.

FakeNewsNet, presented by Shu et al. (2020), serves as a valuable repository of fake news data. This resource provides datasets with diverse features, spatiotemporal information, and social context, facilitating research in the domain of fake news. Evaluation indicates that user engagements can contribute to fake news detection in addition to news articles, highlighting the multifaceted nature of information dissemination.

Kumar et al. (2020) proposed a CNN and bidirectional LSTM ensembled network for identifying original and false news instances. Utilizing various advanced approaches, such as Long Short Term Memories LSTMs, Convolutional Neural Networks CNNs, attention mechanisms, and ensemble methods, the study collected news instances from sources like PolitiFact. The CNN and bidirectional LSTM ensembled network, incorporating an attention mechanism, demonstrated superior accuracy, emphasizing the significance of model complexity in addressing the fake news identification challenge.

Natural Language Processing (NLP) emerges as a pivotal tool in tackling fake news. Choudhary and Arora (2021) proposed a linguistic model, employing handcrafted linguistic features for fake news detection. The model, driven by language-specific features, demonstrated a remarkable 86% accuracy in detecting and categorizing fake messages. Additionally, Abdullah et al. (2020) adopted a multimodal approach, combining Convolutional Neural Network (CNN) and

Long Short-Term Memory (LSTM), achieving significant performance in classifying fake news articles based on source, history, and linguistic cues.

Furthermore, Aslam et al. (2021) introduced an ensemble-based deep learning model for classifying news as fake or real using the LIAR dataset. Employing a combination of Bi-LSTM-GRU-dense and dense deep learning models, the study achieved notable accuracy, recall, precision, and F-score. Despite these advancements, ongoing research aims to enhance the robustness of these models, emphasizing the need for continual improvement and exploration of diverse datasets in fake news detection.

2.1 Evaluation of Existing Scholarship on Fake News Detection

The comprehensive examination of the literature reveals a multifaceted landscape in the realm of fake news identification. Researchers employ diverse strategies, ranging from linguistic models to multimodal approaches, emphasizing the need for a holistic understanding that incorporates both content and social context. Key findings underscore the significance of model complexity, with advanced architectures like the CNN bidirectional LSTM ensembled network exhibiting notable success. Natural Language Processing (NLP) emerges as a critical tool in deciphering news content, demonstrated by linguistic models and the utilization of NLP techniques for textual attribute analysis. Despite considerable progress, there remains a persistent call for improvement, urging researchers to explore feature richness, latent semantic features, and diverse datasets. The global impact of infodemics, particularly highlighted during the COVID-19 outbreak, underscores the urgency in developing robust fake news detection systems. In essence, the literature review illuminates the dynamic and evolving nature of fake news research, emphasizing innovation and adaptability in response to the challenges presented in the digital information age.

2.2 Summary

In summary, this literature review provides a comprehensive exploration of the current state of research in the field of fake news detection. The chapter commences by delineating the landscape of social media and its role in the dissemination of misinformation. It delves into the inadvertent factors contributing to the emergence of fake news, exemplifying instances such as the Nepal Earthquake case. The review accentuates the gravity of the 'infodemic,' particularly evident during the COVID-19 outbreak, necessitating advanced detection mechanisms. Several notable research endeavors are scrutinized, including Sahoo et al.'s automatic fake news identification technique, Shu et al.'s FakeNewsNet repository, and Kumar et al.'s CNN+bidirectional LSTM ensembled network. The significance of Natural Language Processing (NLP) in understanding and detecting fake news is highlighted through studies like Choudhari et al.'s linguistic model. The literature underscores the need for continuous innovation, feature exploration, and adaptation to address the evolving challenges posed by the rampant spread of fake news. This synthesis of existing knowledge provides a robust foundation for the ensuing research endeavors aimed at enhancing the efficacy of fake news detection systems.

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 $x_i \in \mathbf{x}$:

$$s = \frac{1}{N} \sum_{i=1}^{N} x_i. \tag{3.1}$$

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.

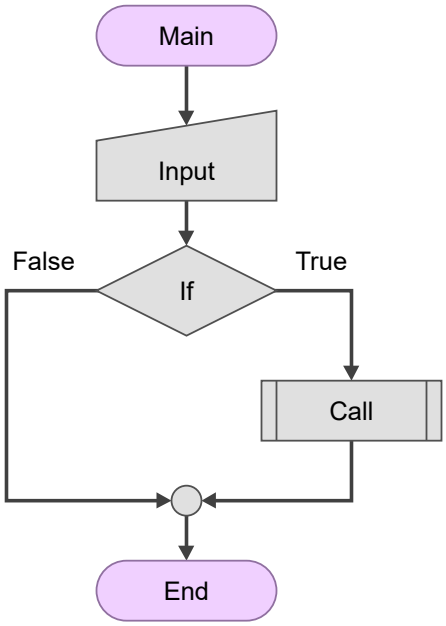

Figure 3.1: Example figure in LaTeX.

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
import numpy as np

x  = [0, 1, 2, 3, 4, 5] # assign values to an array
evenSum = evenSummation(x) # call a function

def evenSummation(x):
    evenSum = 0
    n = len(x)
    for i in range(n):
        if np.mod(x[i],2) == 0: # check if a number is even?
            evenSum = evenSum + x[i]
    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.

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

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

```cpp
int evenSummation(int x[]){
    int evenSum = 0;
    int n = sizeof(x);
    for(int i = 0; i < n; i++){
        if(x[i]%2 == 0){ // check if a number is even?
            evenSum = evenSum + x[i];
      }
    }
    return evenSum;
}

int main(){
    int x[]  = {0, 1, 2, 3, 4, 5}; // assign values to an array
    int evenSum = evenSummation(x);
    cout<<evenSum;
    return 0;
}
```

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

Abdullah, A., Awan, M., Shehzad, M. and Ashraf, M. (2020), 'Fake news classification bimodal using convolutional neural network and long short-term memory', *Int. J. Emerg. Technol. Learn* **11**, 209–212.

Aslam, N., Ullah Khan, I., Alotaibi, F. S., Aldaej, L. A. and Aldubaikil, A. K. (2021), 'Fake detect: A deep learning ensemble model for fake news detection', *complexity* **2021**, 1–8.

Choudhary, A. and Arora, A. (2021), 'Linguistic feature based learning model for fake news detection and classification', *Expert Systems with Applications* **169**, 114171.

Gil, P. (2019), 'What exactly is twitter?'.
URL: *https://www.lifewire.com/what-exactly-is-twitter-2483331*

Kumar, S., Asthana, R., Upadhyay, S., Upreti, N. and Akbar, M. (2020), 'Fake news detection using deep learning models: A novel approach', **31**(2), e3767. e3767 ETT-19-0216.R1.
URL: *https://onlinelibrary.wiley.com/doi/abs/10.1002/ett.3767*

Radianti, J. et al. (2016), An overview of public concerns during the recovery period after a major earthquake: Nepal twitter analysis, *in* 'HICSS '16 Proceedings of the 2016 49th Hawaii International Conference on System Sciences (HICSS)', IEEE, pp. 136–145.

Sahoo, S. R. and Gupta, B. B. (2021), 'Multiple features based approach for automatic fake news detection on social networks using deep learning', *Applied Soft Computing* **100**, 106983.

Shu, K., Mahudeswaran, D., Wang, S., Lee, D. and Liu, H. (2020), 'Fakenewsnet: A data repository with news content, social context, and spatiotemporal information for studying fake news on social media', *Big data* **8**(3), 171–188.

Tandoc Jr, E. C. et al. (2017), 'Defining fake news: A typology of scholarly definitions', *Digital Journalism* pp. 1–17.

Using Social Media (n.d.), `https://esrc.ukri.org/research/impact-toolkit/social-media/using-social-media`. Economic and Social Research Council.

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)

...