CYBER PARENTAL CONTROL FRAMEWORK FOR OBJECTIONABLE WEB CONTENT CLASSIFICATION AND FILTERING BASED ON TOPIC MODELLING USING ENHANCED LATENT DIRICHLET ALLOCATION

HAMZA H. M. ALTARTURI

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY
UNIVERSITI MALAYA
KUALA LUMPUR

2023

CYBER PARENTAL CONTROL FRAMEWORK FOR OBJECTIONABLE WEB CONTENT CLASSIFICATION AND FILTERING BASED ON TOPIC MODELLING USING ENHANCED LATENT DIRICHLET ALLOCATION

HAMZA H. M. ALTARTURI

THESIS SUBMITTED IN FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY
UNIVERSITI MALAYA
KUALA LUMPUR

2023

UNIVERSITY OF MALAYA

ORIGINAL LITERARY WORK DECLARATION

Name of Candidate: Hamza H. M. Altarturi
Matric No: 17050888 | WVA170037
Name of Degree: Doctor of Philosophy
Title of Project Thesis ("this Work"): CYBER PARENTAL CONTROL FRAMEWORK FOR OBJECTIONABLE WEB CONTENT CLASSIFICATION AND FILTERING BASED ON TOPIC MODELLING USING ENHANCED LATENT DIRICHLET ALLOCATION

Field of Study: Cyber Security

I do solemnly and sincerely declare that:

(1) I am the sole author/writer of this Work;
(2) This Work is original;
(3) Any use of any work in which copyright exists was done by way of fair dealing and for permitted purposes and any excerpt or extract from, or reference to or reproduction of any copyright work has been disclosed expressly and sufficiently and the title of the Work and its authorship have been acknowledged in this Work;
(4) I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;
(5) I hereby assign all and every rights in the copyright to this Work to the University of Malaya ("UM"), who henceforth shall be owner of the copyright in this Work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of UM having been first had and obtained;
(6) I am fully aware that if in the course of making this Work I have infringed any copyright whether intentionally or otherwise, I may be subject to legal action or any other action as may be determined by UM.

Candidate's Signature Date: 1/11/2023

Subscribed and solemnly declared before,

Witness's Signature Date: 1/11/2023

Name:

Designation:

CYBER PARENTAL CONTROL FRAMEWORK FOR OBJECTIONABLE WEB CONTENT CLASSIFICATION AND FILTERING BASED ON TOPIC MODELLING USING ENHANCED LATENT DIRICHLET ALLOCATION

ABSTRACT

The escalating concern revolves around cybersecurity for children, given the unprecedented internet access that potentially exposes them to objectionable content. Recent data highlight the problem's severity, revealing a 97% surge in children's online exploitation and a 28% rise in reported minor sexual abuse material online. This problem has motivated academia and industry to develop frameworks for cyber parental control. Despite substantial advancements in automating web classification that combines web mining and content classification methods, the study identifies a gap in applying advanced machine learning algorithms for superior objectionable web content classification. Most existing studies adopt one classifying approach, resulting in an ineffective and unreliable classification of objectionable content. In terms of content, only a few studies address a wide range of objectionable content topics, whereas most studies primarily focus on pornography topics. Furthermore, studies on classifying objectionable contents use conventional topic models, such as the Latent Dirichlet Allocation (LDA) and its variants. These models are built to work on generic fields and conventional documents, ignoring the structure of web content in the HTML documents and insufficiently performing when applied to web content data. Neglecting the unique structure of web content leads to missing the otherwise interpretable topics and, therefore, to low topic quality and classification accuracy. Moreover, the lack of publicly accessible objectionable web content ground-truth datasets has prevented a fair, coherent comparison of the various frameworks. This research aims to propose an effective and accurate framework for classifying objectionable web content. The Cyber Parental Control Framework (CPCF) employs a multistep approach and a novel web mining technique. It uses the URL blacklist and whitelist methods as the first and second filter layers. A final classification

layer is then applied in which an HTML Topic Model (HTM) developed by this study analyses HTML tags to understand the structure of the webpages. The HTM is an enhancement of the LDA model. This study creates a ground-truth objectionable web content dataset to achieve the aim. The ground-truth dataset contains 8,000 labelled websites, split equally between objectionable and unobjectionable websites and comprising over 2 million pages. The study conducted four series of experiments to examine the CPCF. The first experiment's results demonstrate the reliability of the ground-truth dataset using the existing state-of-the-art classifiers. The results of the second experiment demonstrate the limitations of the existing topic models web applied to web content. The third experiment then evaluates the effectiveness of the HTM in discovering interpretable topics and term patterns compared to the widely used LDA model. The final experiment investigates the performance and accuracy of the CPCF using the HTM model. The CPCF demonstrates effectiveness in web content classification and the ability to overcome the limitations of the existing methods. Finally, a web-based functional prototype was developed to facilitate the CPCF's applicability and to offer a valuable reference for future research and prospects in this domain. The contribution of this study is a framework to produce an objectionable web content classification for cyber parental control, which was proposed, designed, evaluated, and simulated.

Keywords: Machine Learning, Natural Language Processing, Latent Dirichlet Allocation, Topic Modeling, Language Model, Cyber Security, Web Classification.

RANGKA KERJA KAWALAN IBU BAPA SIBER UNTUK KLASIFIKASI KANDUNGAN WEB BOLEH DIBANTAH DAN PENAPIS BERDASARKAN PEMODELAN TOPIK MENGGUNAKAN PERUNTUKAN DIRICHLET TERDAM YANG DIPERTINGKAT

ABSTRAK

Kebolehcapaian Internet yang semakin meningkat dari pelbagai tempat dan peranti menjadikan banyak kandungan web, termasuk kandungan yang kurang sesuai tersedia kepada pengguna. Kandungan yang kurang sesuai seperti pornografi, dadah, senjata, perjudian, keganasan dan kebencian, menimbulkan masalah serius bagi pengguna Internet, terutamanya kanak-kanak. Kira-kira 65% kanak-kanak di UK telah secara tidak sengaja melihat kandungan yang kurang sesuai di Internet. Masalah ini telah mendorong ahli akademik dan industri untuk membangunkan kaedah kawalan siber bagi ibu bapa serta klasifikasi dan penapisan kandungan web yang kurang sesuai ini. Nilai pasaran kawalan siber bagu ibu bapa di peringkat global dianggarkan berjumlah USD 1,400 juta pada 2016 dan dijangka mencecah USD 3,300 juta menjelang 2025. Beberapa kajian penyelidikan telah membentangkan pendekatan klasifikasi web automatik yang menggabungkan perlombongan web dan klasifikasi kandungan dan teknik penapisan dengan beberapa kaedah , seperti kaedah berasaskan URL, kata kunci, kandungan dan Platform untuk Pemilihan Kandungan Internet (PICS). Kebanyakan kajian ini hanya menggunakan satu pendekatan pengelasan sekaligus menyebabkan pengelasan dan penapisan kandungan yang kurang berkesan dan tidak boleh dipercayai. Dari segi kandungan, hanya terdapat beberapa kajian yang menangani pelbagai topik kandungan yang kurang sesuai, manakala kebanyakan kajian lain memfokuskan pada mengklasifikasikan pornografi dan topik yang berkaitan. Tambahan pula, kajian mengenai pengkelasan kandungan yang kurang sesuai menggunakan model topik

konvensional, seperti model *Latent Dirichlet Allocation* dan variannya. Model ini dibina untuk berfungsi pada medan generik dan dokumen konvensional sekaligus mengabaikan struktur kandungan web dalam dokumen HTML dan seterusnya menurunkan prestasi apabila digunakan pada data kandungan web. Pengabaian struktur unik kandungan web membawa kepada kehilangan topik yang boleh ditafsir dan, oleh itu, kepada kualiti topik dan ketepatan klasifikasi yang rendah. Selain itu, kekurangan set data penanda aras berkaitan kandungan web yang boleh diakses secara terbuka telah menghalang perbandingan yang adil bagi pelbagai rangka kerja yang telah dicadangkan dalam bidang kawalan siber bagi ibu bapa. Oleh itu, kajian ini bertujuan untuk mencadangkan rangka kerja yang berkesan dan tepat untuk mengklasifikasikan kandungan web yang kurang sesuai. Rangka kerja ini menggunakan pendekatan berbilang langkah dan teknik perlombongan web yang baru. Ia menggunakan kaedah senarai hitam dan senarai putih URL sebagai lapisan penapis pertama dan kedua. Lapisan klasifikasi akhir kemudiannya digunakan di mana *HTML Topic Model* (HTM) yang dibangunkan dalam kajian ini menganalisis tag HTML untuk memahami struktur halaman web. Model HTM adalah berasaskan daripada penambah baikan kepada model *Latent Dirichlet Allocation*. Kajian ini turut membangunkan set data penanda aras berkaitan kandungan web yang kurang sesuai . Set data ini mengandungi 8,000 laman web berlabel, dibahagikan sama rata antara laman web yang kurang sesuai dan yang sesuai serta terdiri daripada lebih 2 juta halaman web. Kajian ini telah menjalankan empat siri eksperimen untuk mengkaji rangka kerja yang dicadangkan untuk kawalan siber bagi ibu bapa. Keputusan eksperimen pertama menunjukkan kebolehpercayaan set data penanda aras menggunakan pengelas terkini yang sedia ada. Keputusan eksperimen kedua menunjukkan had model topik sedia ada yang digunakan web pada kandungan web. Eksperimen ketiga kemudiannya menilai keberkesanan model HTM yang dicadangkan dalam menemui topik yang boleh ditafsir dan corak istilah dalam data kandungan web berbanding model topik LDA yang

digunakan secara meluas. Eksperimen akhir menyiasat prestasi dan ketepatan rangka kerja kawalan siber bagi ibu bapa menggunakan model HTM yang dicadangkan. Rangka kerja yang dicadangkan menunjukkan keberkesanan dalam klasifikasi kandungan web dan keupayaan untuk mengatasi batasan kaedah sedia ada. Akhirnya, sebuah prototaip berasaskan web telah dibangunkan untuk memudahkan penggunaan rangka kerja yang dicadangkan untuk kawalan siber bagi bapa dan untuk menawarkan rujukan berharga bagi penyelidikan masa depan dalam bidang ini. Sumbangan kajian ini adalah mencadangkan, mereka bentuk, menilai dan mensimulasikan rangka kerja untuk menghasilkan klasifikasi dan penapisan kandungan web yang kurang sesuai untuk kawalan siber bagi ibu bapa.

Kata kunci: Kawalan siber bagi ibu bapa, klasifikasi web, perlombongan kandungan web, pemodelan topik, pembelajaran mesin, *Latent Dirichlet Allocation*.

ACKNOWLEDGEMENT

And my success is not but through Allah (The Qur'an,11:88). My deepest gratitude to the almighty Allah, who has always blessed me and endowed me with the determination to complete my PhD research journey.

First and foremost, I would like to express my sincere gratitude to my supervisor and advisor, Prof. Dr. Nor Badrul Anuar Bin Juma'at, for his continuous support in my research journey. His profound knowledge and invaluable guidance have been instrumental to my academic growth. I will always be indebted to his commitment, which not only taught me the fundamentals of my research domain but also provided the blueprint for scientific reasoning, discipline, and resilience.

I am forever and deeply thankful to my beloved mother, Ustazah Jamilah Sulaiman Moheisen, and my finest father, Prof. Dr. Hussein Motawe Altarturi, for their unconditional love and endless support and for instilling in me the passion for learning. To my siblings, Ala, Anas, Omama, Isra, Bara, Basheer, and Hanan, thank you for your constant encouragement and unwavering belief in my capabilities. I love you all.

Special thanks go to my friends, Ahmad, Muntadher, Hussein, Haqi, and Hasan, for being my family away from home. Their continued support, friendly advice, and endless cups of coffee have made the PhD journey a memorable experience.

I gratefully acknowledge the financial support from the Universiti Malaya Research Grant, Fundamental Research Grant Scheme, and Malaysia International Scholarship. These funds have provided me with the opportunity to concentrate on my research without financial burdens.

To all those who contributed directly or indirectly to this journey, please know that your impact is truly appreciated and will not be forgotten.

In these challenging times, my heart constantly prays for the freedom and peace of my homeland, PALESTINE, and I yearn for the day when its people will live without oppression.

Last but not least, I want to thank me, for so many reasons...

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS AND ABBREVIATIONS

PICS	:	Platform for Internet Content Selection
VSM	:	Vector Space Model
LDA	:	Latent Dirichlet Allocation
pLSA	:	Probabilistic Latent Semantic Indexing
HTM	:	HTML Topic Model
SVM	:	Support Vector Machine
W3C	:	World Wide Web Consortium
POWDER	:	Protocol for Web Description Resources
ICA	:	Intelligent Content Analysis
NLP	:	Natural Language Processing
WSD	:	Word Sense Disambiguation
SSO	:	Simplified Swarm Optimisation
SEO	:	Search Engine Optimisation
NB	:	Naïve Bayes
RF	:	Random Forest
LR	:	Logistic Regression
KNN	:	K-Nearest Neighbor
CNN	:	Convolutional Neural Network
KL	:	Kullback-Leibler
DP	:	Dirichlet Process
PYP	:	Pitman-Yor Process
LSI	:	Latent Semantic Indexing
pLSI	:	Probabilistic Latent Semantic Indexing
CTM	:	Correlation Topic Model
PAM	:	Pachinko Allocation Model
sLDA	:	Supervised Latent Dirichlet Allocation
LLDA	:	Label-Latent Dirichlet Allocation
DMR	:	Dirichlet-Multinomial Regression
HDP	:	Hierarchical Dirichlet Process
HLDA	:	Hierarchical Latent Dirichlet Allocation
HPYP	:	Hierarchical Pitman-Yor Process
PTM	:	Pseudo-document-based Topic Model

LSA	:	Latent Semantic Analysis
NLTK	:	Natural Language Toolkit
DOM	:	Document Object Model
DMOZ		Directory Mozilla
id2word	:	Word Identification
BoW	:	Bag of Words
CD-based	:	Conventional Document-based
WC-based	:	Web Content-based
HTML	:	HyperText Markup Language
TLD	:	Top-Level Domain
NPMI	:	Normalized Pointwise Mutual Information
PMI	:	Pointwise Mutual Information
TP	:	True Positive
TN	:	True Negative
FP	:	False Positive
FN	:	False Negative
API	:	Application Programming Interface
ML-HTM	:	Multi-Lingual HTML Topic Model

LIST OF APPENDICES

CHAPTER 1: INTRODUCTION

This chapter introduces the big picture of this study. It contains a general background and the topics related to this study (Section 1.1), followed by the motivation of this study (Section 1.2). Section 1.3 addresses the problem statement, while Section 1.4 defines the research aim and objectives of the study. Section 1.5 briefly explains the methodology of this study. Finally, Section 1.6 presents the structure of the following chapters.

1.1. Research Background

The growing Internet accessibility from various places and devices makes a vast amount of web content, including objectionable materials, available to users. These objectionable contents, such as pornography, drugs, weapons, gambling, violence, and hatred, pose serious problems for Internet users, especially children. This problem has motivated academia and industry to develop cyber parental control tools to protect children when using the Internet.

Cyber parental control, a subfield of cyber security, is related to a few subjects; parental monitoring, mediation, and control.

a) *Parental monitoring*. (Dishion & McMahon, 1998) define parental monitoring as "a set of correlated parenting behaviours involving attention to and tracking of the child's whereabouts, activities, and adaptations". The definition includes two important aspects of parental monitoring; actively controlling and keeping watch over the children (Law et al., 2010). This definition intersects with the definition of parental mediation; indeed, some researchers refer to parental mediation as parental media monitoring (Padilla-Walker et al., 2012).
b) *Parental mediation*. Implicates interactions between parents or guardians and their children over the media and includes (a) restrictive mediation (including limiting and controlling children's Internet activities); (b) evaluative mediation (including an open

discussion concerning the Internet and joint creation of rules); and (c) co-using (including parents' active participation with children's online use, including recommending websites and participating in online activities) (Elsaesser et al., 2017; Valkenburg et al., 1999).

c) *Parental control*. In general, this refers to allowing parents or children's guardians to know and control whom their children interact with, where their children are, and what they do inside and outside the house through rules and restrictions (Aunola et al., 2015).

Taken together, this study conceptualises the cyber parental control term as a collection of parenting actions involving monitoring, controlling, and limiting children's online activities. Figure 1.1 illustrates the relationship between cyber parental control and parental monitoring and mediation. It also shows that cyber parental control includes cyber monitoring, controlling, and filtering:

a) *Cyber Monitoring*. Focuses on observing and detecting potential threats or unauthorized activities.
b) *Cyber Controlling*. Concerned with managing and regulating access and operations within the system.
c) *Cyber Filtering*. Aims to classify and filter objectionable or harmful content from being accessed within a network or system.

Cyber monitoring and controlling, while essential in the broader context of cybersecurity, neglect the classification and filtering of objectionable web content, while cyber filtering directly aligns with the study's aim to classify and filter objectionable web content (Deibert et al., 2010; Ding et al., 2019).

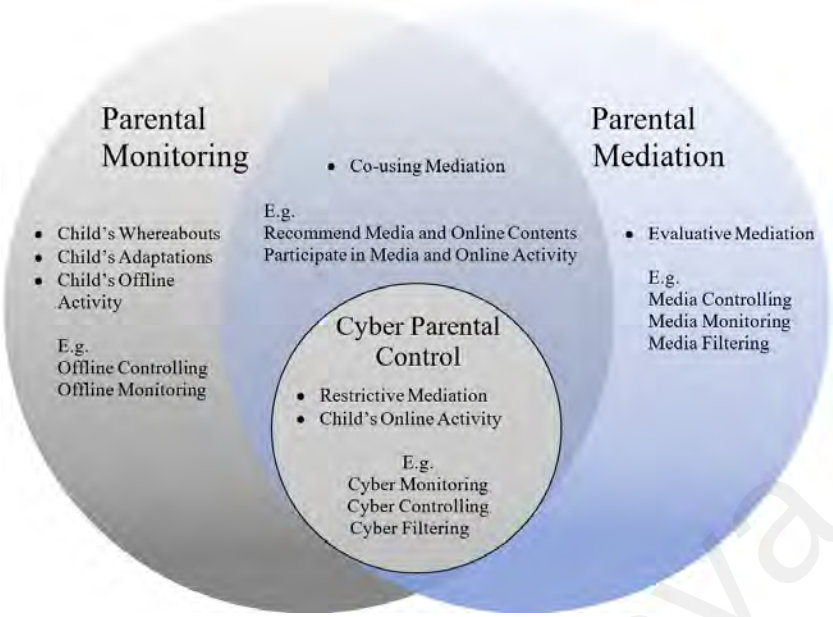

Figure 1.1: Cyber parental control concept

There are two main ways for cyber filtering: manually and automatically. Cyber filtering, also known as web filtering, manually by an expert is nearly impossible because of the sheer number of websites – in January 2018, there were 1,805,260,010 websites (Netcraft, 2018, 19 January), and the number increased daily. For instance, in 2017 only, an average of one hundred thousand websites were added every day (Netcraft, 2018, 19 January). Automatic filtering uses methods and techniques of web mining to classify and filter websites, and it comes in three categories; web structure mining, web usage mining, and web content mining (R. Kosala & H. Blockeel, 2000; Lee et al., 2015).

Classifying and filtering web content includes four approaches: URL-based, keyword-based, content-based, and Platform for Internet Content Selection (PICS)-based. Since the PICS-based and keyword-based approaches are fast and lightweight (which leads to an efficient, easy, and low-cost resource filtering framework (Ahmed & Jameel, 2022; Minh et al., 2022)), the majority of previous studies focus on them. Keyword-based policies depend on the list of references and fail mostly because of the incompleteness of these references. A few other types of research use the PICS method that uses metadata to determine the scope of web pages and sites. Metadata plays an essential role in this

mechanism, which some companies intentionally misname to overcome the filter. Adopting solely an approach that performs filtering based on the keywords or PICS, therefore, results in ineffective and unreliable filtering of objectionable content for several reasons (Lee et al., 2003; Bhavish Khanna Narayanan et al., 2018; Zeng et al., 2013), explained as follows:

a) *Over-blocking*. This issue prevents access to important information and resources that are wrongly filtered. Filtering systems that are based on URL and PICS-based approaches suffer from this significant issue (Khan et al., 2021).
b) *Under-blocking*. Unlike over-blocking, this issue results in failing to block all inappropriate content. With the massive increase in website numbers (Netcraft, 2018), URL and PICS-based filtering systems fail to keep their reference lists up to date.
c) *Context Ignorance*. While the URL and PICS approaches count out the context of the web content, the keyword-based approach often struggles with understanding the context in which words are used (Hariri et al., 2019).
d) *Language and Slang*. Slang terms can change rapidly and often vary by region or subculture (MacAvaney et al., 2019), which can challenge keyword-based filtering systems to filter inappropriate web pages.

The inefficiency and unreliability of filtering objectionable web content topics resulting from these problems raise the importance and need for an efficient web topic model to classify and filter web content.

Researchers have designed several topic models, some of which are generic and comprehensive, such as the Latent Dirichlet Allocation (LDA) (Blei et al., 2003). Other models, in contrast, are designed to work with particular and specific tasks, such as the Time Author Topic model (TAT) (Xu et al., 2014) and the Twitter-LDA (Zhao et al., 2011). Despite their design differences and applications, topic models are Vector Space

Model (VSM) based. Belei et al. (2003) proposed the LDA, which extended the Probabilistic Latent Semantic Indexing (pLSA) by adding Dirichlet priors on topic distributions (Alkhodair et al., 2018; Hajjem & Latiri, 2017). The LDA has proven its solid baseline in generating coherent topics (topics are supported by a text set (called reference corpus)) and has been widely used in various applications, including generative language models, spam filtering, and recommender systems. Several studies followed proposing topic models that add constraints on the traditional LDA to generate modified models called LDA variants (Chen et al., 2012).

Taking all the abovementioned aspects together, this study focuses on filtering objectionable web content based on web content mining using topic modeling. This includes the following topics, as Figure 1.2 illustrates:

a) *Parental control* (addressed in Chapter 1)

b) *Objectionable content* (addressed in Chapter 1)

c) *Web mining* (addressed in Chapter 2)

d) *Web filtering* (addressed in Chapter 2)

e) *Topic modeling* (addressed in Chapter 3)

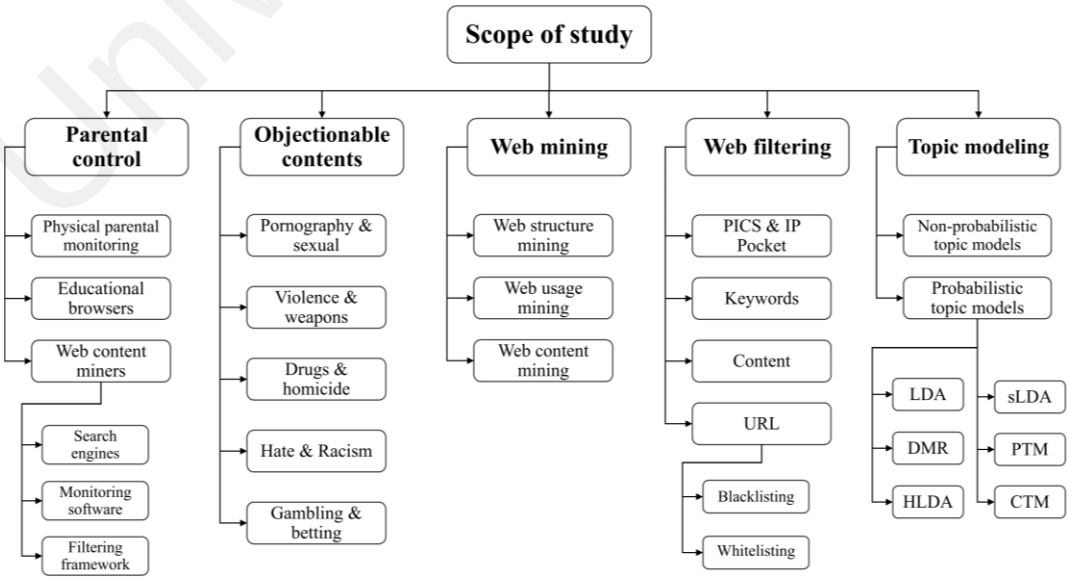

Figure 1.2: Scope of this study

1.2. Research Motivation

The Internet has become the main place for children's education and social interaction, and this study's significant motivation is to provide them with a safe online environment without them being exposed to harmful web content. The study of cyber parental control holds significant importance due to the extensive use of the Internet by a massive number of children, which is inundated with objectionable content. For instance, in Malaysia, Internet users in 2014 were about 24.5 million, 15% of whom were children, and the percentage has been increasing since then (MCMC, 2017; UNICEF Malaysia & Digi, 2015). The Internet plays an essential role in children's education, entertainment, and socialisation. Internet includes, however, objectionable content, such as pornography, drugs, weapons, gambling, violence, hatred, and bullying. These objectionable contents pose serious problems and risks for Internet users, especially children. About 65% of children in the UK have seen, mostly accidentally, objectionable content on the Internet (Martellozzo et al., 2016). These online risks require parents to use cyber parental control tools to protect children when using the Internet.

From an economic perspective, the value of the global parental control market was estimated at USD 1,400 million in 2016 and is expected to reach USD 3,300 million by 2025 (Research, 2018). The report expects the global parental control market to exhibit a Compound Annual Growth Rate (CAGR) of over 11.5% between 2017 and 2025. For instance, VP Capital and Larnabel Ventures announced a $2 Million investment in Smart Parental Control Startup FaceMetrics (Capital, 2018). Academia has also been studying the field of cyber parental control from an academic aspect.

Failing to correctly classify web content and detect objectionable topics means potentially harmful material finds its way to children. Extracting coherent web content topics is significant in filtering all objectionable content from the web and providing a

safe Internet environment for children. However, existing topic models fail to generate coherent topics from web content (Altarturi et al., 2023).

The drawback of the existing topic models on web content data raises the need for a framework for objectionable web content classification to provide a safe Internet for children. By exploiting the better means of topic modeling approaches, the framework filters the web pages based on their objectionability content.

1.3. Problem Statement

Internet access has become possible nowadays from all mobile devices, making web content, including objectionable materials, available to users nearly everywhere. Classification of content based on its suitability for children is necessary to enable blocking objectionable content, thus providing a safe environment for that colossal number of Internet children users (Altarturi et al., 2020). Although research in this area started more than two decades ago, it became more significant because of the dramatic growth in the availability of information resources on the web (AlAgha, 2022; Artene et al., 2022; Berardi et al., 2015; Yenala et al., 2018). Classifying objectionable content, however, is nearly impossible for an individual with traditional methods due to the sheer amount of content on the Internet. Topic modeling, a text-mining tool, aims to discover latent semantic structures or topics within a set of textual digital documents.

The majority of previous studies adopted solely one filtering approach, which results in ineffective and unreliable filtering of objectionable content (Demirkıran et al., 2020; B. K. Narayanan et al., 2018; Zeng et al., 2013). All objectionable contents, not limited to pornography, can negatively affect children psychologically and mentally. However, only a few studies have addressed a wide range of objectionable topics. These studies use conventional topic models that ignore web content structure, resulting in uncoherent topic

generating and, therefore, ineffective and inaccurate filtering of objectionable web content.

1.4. Research Objectives

As the problem statement section discussed, the current methods for objectionable web content classifying and filtering present significant issues. To address these issues, this study aims to develop an effective and accurate framework for classifying and filtering objectionable web content. This aim is achieved by proposing a coherent web content topic modeling based on novel web mining techniques. The main objectives of this study are as follows:

1. To analyse the existing web content filtering and topic modeling approaches used in cyber parental control.
2. To design a coherent topic model for learning coherent topics in web content data.
3. To develop a web content classifying and filtering framework based on the proposed topic model, whitelisting URLs, and blacklisting URLs.
4. To evaluate the performance of the developed framework in terms of topic coherence and classification effectiveness and accuracy.

Given the main goal of providing a safe Internet environment for children, this study centers on the objectionability of topics considering children and cyber parental control. The general principle and proposed framework, however, apply to all other web content topics.

The objectives presented above describe the overall sequence of the material and the structure presented in this study.

1.5. Research Methodology

This study is decomposed into four phases; literature review and problem identification, model design, framework development, and validation and evaluation as follows:

a) *Phase 1*. The initial phase of the adapted methodology is the literature review, which aims to fulfill the first objective of this study: to analyse the existing web content filtering and topic modeling approaches used in cyber parental control. It answers the following questions by reviewing the state-of-the-art literature and bibliography:

- What is cyber parental control?
- What are topic modeling approaches for web content?
- What are the available and suitable ground truth datasets for objectionable content?
- What challenges hinder proposing an efficient objectionable content filtering for the web using topic modeling?

The outcome of this phase delineates the landscape of cyber parental control and topic modeling methods for web content classification and outlines key challenges that necessitate further research to enhance web content classification efficacy.

b) *Phase 2*. This phase includes designing and developing a coherent topic model for web content. The designed topic model takes into consideration the structure of the web (due to its importance to understand the content of a webpage) to learn coherent topics and, therefore, fulfills the second objective of this study. This phase provides answers to the following questions by conducting a comprehensive comparison of the benchmark topic models and mathematical designs:

- What are the strengths and weaknesses of the existing topic models?
- What is the robust mathematical model to be adopted for the web content topic model?

- How to take into consideration the webpage structure in the mathematical representation of the topic model?

The outcome of this phase is a novel topic model design for web content that utilizes the web structure to determine coherent topics, achieved by critically analyzing existing models, selecting an optimal mathematical model, and successfully incorporating webpage structure into the model's mathematical representation.

c) *Phase 3*. This phase focuses on designing and developing the cyber parental control framework that filters objectionable web content based on the topics generated by the designed topic model. Finally, it also implements the developed framework as a web-based application. The framework also includes two prior layers to enhance its efficiency; the whitelisting and blacklisting layers. The development of this framework fulfills the third objective of this study and answers the following questions:
   - How to classify and filter objectionable web content data based on the proposed topic model?
   - How to integrate the topic model layer with the whitelisting and blacklisting URL filtering layers?
   - How to develop and implement the component of the integrated framework?

   The outcome of this phase is the developed framework which efficiently integrates the novel topic model as a layer, along with the whitelisting and blacklisting layers, demonstrating a practicable approach to classifying and filtering web content data.

d) *Phase 4*. The final phase aims to evaluate the topic coherence of the proposed topic model and the effectiveness and accuracy of the developed cyber parental control framework based on the generated topics, fulfilling the fourth objective of the study.

The validation and evaluation phase answers the following questions by conducting a series of experiments using several metrics:

- What is the coherence of the proposed topic model?
- How to design and develop a ground truth dataset for objectionable content?
- What is the accuracy of the proposed framework in classifying and filtering objectionable topics?

The output of this phase is an evaluated cyber parental control framework through a series of experimental metrics, exhibiting high coherency in generating topics and effectiveness and accuracy in classifying and filtering objectionable topics.

Figure 1.3 illustrates the research methodology and study phases interlinked with the research objectives and questions.

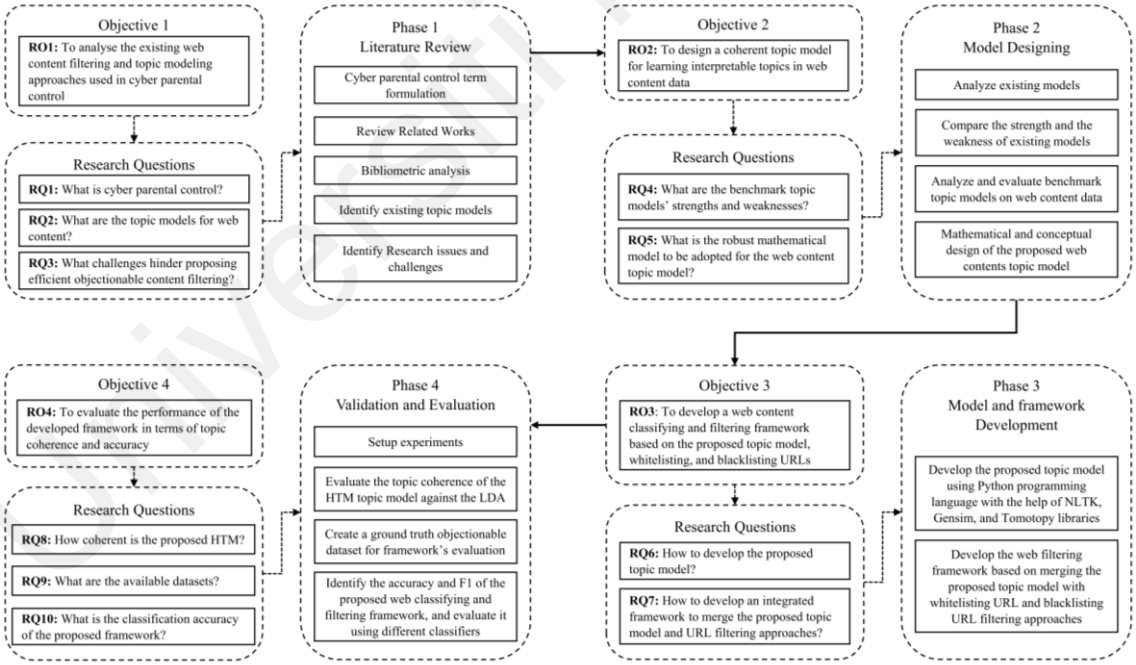

Figure 1.3: Methodology of the study

1.6. Thesis Structure

Chapter 2 introduces the domain of cyber parental control and its state-of-the-art, highlighting the challenges in ensuring a safe online environment for children due to easily accessible objectionable web content. This chapter comprehensively reviews

current methods and techniques for classifying and filtering web content. It begins by explaining the concept of cyber parental control and its associated facets, then delves into the techniques used for web content categorisation and parental control frameworks. Subsequently, it reviews state-of-the-art filtering methods, including the web mining techniques applied in the study. Despite substantial advancements in content filtering and web content mining techniques, the chapter identifies a gap in applying advanced machine learning algorithms for superior web content filtering. The next chapter discusses topic models and their web applications, aiming to fill this gap with topic modeling for improved classification and filtering of objectionable content.

Chapter 3 explores the utility of topic modeling, a technique for discovering latent semantic structures in large volumes of textual digital documents in web applications. Despite the widespread use of topic models, their application in web content classification and filtering remains underexplored. This chapter provides a comprehensive overview of the mathematical foundations of topic modeling and synthesizes state-of-the-art topic models into a taxonomy. Scrutinizing this taxonomy reveals the need for a more coherent topic model specifically designed for web content data, as the most current models overlook the structure of textual contents within HyperText Markup Language (HTML) tags. The chapter establishes several key findings that set the groundwork for designing and developing an effective and accurate web classification framework using a coherent topic model, discussed in detail in the subsequent chapter.

Chapter 4 presents the core contribution of this study, providing detailed insights into the proposed HTML Topic Model (HTM) and the proposed web classification framework for cyber parental control. The HTM model design enhances topic modeling performance for web content-based data. The classification framework consists of a multistep approach with a three-layer design: the first two layers are URL whitelist and blacklist filters, while the final layer utilizes the HTM for content-based classification. This layer includes

modules for webpage scraping, preprocessing, topic modeling using the HTM, and final classification into objectionable or unobjectionable categories. The chapter then highlights the operational characteristics of this innovative cyber parental control framework. The coherency of the proposed HTM model and the effectiveness and accuracy of the proposed framework are thoroughly evaluated and benchmarked in the subsequent chapter.

Chapter 5 thoroughly evaluates the proposed cyber parental control framework and the HTM using this study's developed ground truth dataset. This chapter presents four progressively conducted experiments to assess the framework's effectiveness and accuracy and to benchmark the HTM topic model against widely used models in the existing literature, including:

a) *Experiment I*. Verifies the ground truth dataset's reliability.
b) *Experiment II*. Uncovers the performance drawbacks of conventional topic models when applied to web content.
c) *Experiment III*. Reveals the HTM's superior performance, demonstrating a 36.5% improvement in topic coherence compared to the LDA model.
d) *Experiment IV*. Employing a ground truth dataset of approximately 2 million web pages. It shows the effectiveness of the proposed cyber parental control framework, achieving an impressive accuracy of 95% when using the proposed HTM topic model, signifying a 30% improvement over conventional topic models.

Chapter 6 details the development and deployment of a web-based prototype implementing the proposed cyber parental control framework. The distributed architecture of the prototype integrates key components, including URL listing layers, web scrapper, pre-processor, HTM topic model, and classifier. These integral components contribute to an abstracted, user-friendly application interface that simplifies the

complexity of the underlying framework. The chapter demonstrates the working prototype, discusses its benefits and limitations, and effectively translates theoretical development into real-world application. Further discussion on the study's contributions, challenges, and future research scope will follow in the next chapter.

Chapter 7 provides a conclusion to the study, summarizing its findings and contributions to the field of cyber parental control and web content classification. The chapter emphasizes the significance of developing an effective classification framework for objectionable web content. This study has successfully designed and implemented such a framework, including the novel HTM, to address the limitations of traditional models that overlook the unique structure of web content. The study further facilitates fair comparisons in future research in this domain by establishing a ground truth dataset. Limitations of the current study and prospective directions for future research are also discussed in this chapter.

CHAPTER 2: OVERVIEW OF CYBER PARENTAL CONTROL AND WEB MINING

The amount of web content is constantly increasing and easily accessible from various places and devices. These contents, including objectionable materials, are available to users and can pose serious problems, especially for children. Consequently, cyber parental control becomes a monumental challenge motivating the innovation of web content classification and filtering. This chapter presents an overview of the state-of-the-art in filtering and classifying objectionable web content. It discusses the concepts of cyber parental control and web mining to lay the foundation for this research. The literature review summarises the essential previous research and their domains, approaches, methods, and algorithms. This chapter aims to synthesise the relevant findings published and reviewed in scientific journals and proceedings and highlights the existing approaches' gaps by comparing the related work.

2.1. Cyber Parental Control

The cyber parental control term is parenting actions involving monitoring, controlling, and limiting children's activities on the cyber. Several factors matter concerning studying cyber parental control, such as sex, age, education, and socioeconomic. Several studies among the analysed dataset have addressed these factors (Wong, 2010). However, the socioeconomic factor was mostly discussed in the conventional parenting control (Jain et al., 2018; Tippett & Wolke, 2014; Top, 2016), while only a few studies investigated the socioeconomic relation with cyber parental control (Ibrahim, 2016a, 2016b).

Although previous studies have investigated the field of cyber parental control from psychological and technological perspectives, few studies address the connection between these perspectives. Investigating this connection is significant for a better understanding of the cyber parental control field. Understanding the field helps propose

a more effective and efficient cyber parental control approach. This study fills in that gap by breaking down the taxonomy of cyber parental control tasks and decomposes the general process of cyber parental control into three tasks, as Figure 2.1 illustrates; (a) define what categories of content on the cyber network need to be controlled; (b) detect the defined contents and identify the required methods and techniques for that; (c) protect and filter the detected contents and identify the required approach for that.

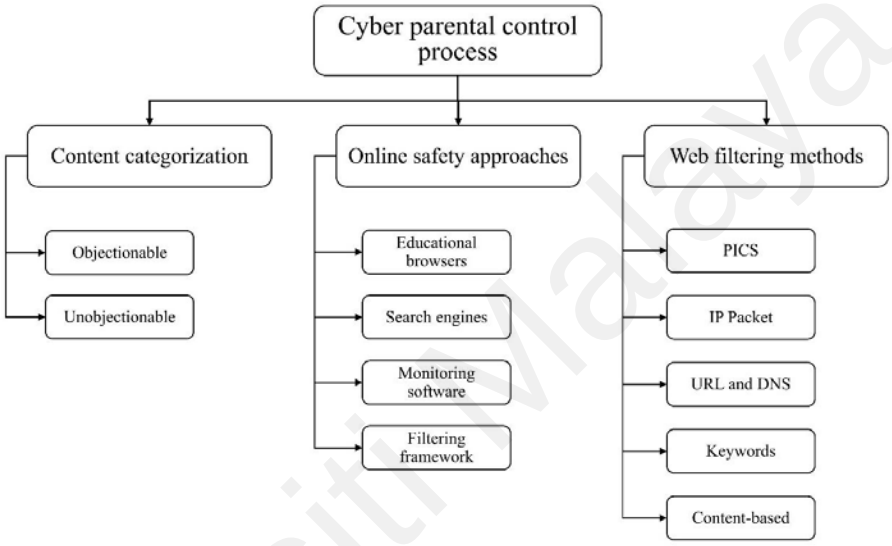

Figure 2.1: The process of cyber parental control

2.1.1. Content Categorisation

Content categorization is classifying content into distinct categories or groups based on shared qualities or topics. It aims to make it easier to locate and access specific sorts of content, evaluate and comprehend the content in a specific context, or filter and classify specific content. There are two significant points in order to categorize web contents:

a) *Criteria of categorizing*. There are numerous ways to classify content, and the categories utilised depend on the nature of the content and the categorization's objectives. The two main methodologies for categorizing web content are content-based and user-based criteria (Almatrooshi et al., 2022).

b) *Type of data*. The web contains several types of content, such as text, image, video, and audio. Regardless of the type, any content contains one or several topics that may be harmful or inappropriate for children and, therefore, objectionable. However, using only textual content efficiently describes all website content types (Stroud et al., 2020).

This study focuses on the topic criteria of categorizing the textual content of the webpage to classify and filter objectionable content. However, there is a lack of definition of objectionable content in the literature. Most of the literature includes pornography and violent topics as objectionable content. Besides these two topics, few studies include drugs, hate, racism, sexual, homicide, gambling, and weapons (Duan & Zeng, 2013; Duan et al., 2012; Hammami et al., 2006; Jacob et al., 2007; Lee et al., 2015; Lee et al., 2003, 2005; Zeng et al., 2013). The linguistic definition of objectionable is anything that certain people dislike or oppose because they are unpleasant or wrong. Considering these points, this study conceptualises the objectionable web content term as textual or visual content that certain internet users oppose on the web, including, but not limited to, pornography, violence, drugs, hate, racism, sexual, homicide, gambling, and weapons.

Although content categorisations are inconsistent across studies and tools, this study elicits a general categorisation from these studies (Eickhoff et al., 2011; Nanny, 2019; Qustodio, 2019; Zeniarja et al., 2018). The adopted categorisation in this study contains two categories of content:

a) *Objectionable and harmful content*. This category contains any topic that causes harm to children mentally, psychologically, or both. Such topics include pornography, racism, drugs, weapons, gambling, and violence.

b) *Unobjectionable and legitimate content*. This category includes any topic that causes no mental or psychological harm to children, merely any other topics not included in the objectionable category.

2.1.2. Online Safety Approaches

The second task of cyber parental control is to detect objectionable content. Literature combines several methods to detect objectionable content, such as classification, categorisation, filtering, and recommendations. There are five approaches to applying these methods and developing cyber parental control tools: educational browsers and tailored browsers, search engines, monitoring software, time control, and filtering software (Hilal & Gupta, 2013). The following sub-sections describe these approaches.

a) *Educational and Tailored Browsers*. It aims to provide a safe browsing environment for children. There are two ways to apply this approach; through an educational browser or add-ons. Educational browsers, such as Kiddle and Kidrex (Kiddle, 2019; kidrex, 2019), were designed to provide a child-friendly and safe interface. Browsers' add-on modules are filtering software attached to various browsers. Users can manually add these add-on modules to their browsers or use the default filtering feature that most browsers, such as Chrome (Google, 2019), provide. Parental control uses this approach to open access to school and educational websites only using a whitelist, blacklist, and keyword filtering approach (Fuertes et al., 2015). The drawback of this approach is preventing access to useful entertainment and social media websites (Hilal & Gupta, 2013). Moreover, children may easily bypass this approach by accessing other websites through different browsers.

b) *Search Engines*. A few studies have built and examined search engines based on their filtering approaches. Many search engines for children are also available on the web, but most use Google's customised search to filter objectionable web content (Dilip Patel & Pandya, 2017). A drawback of this approach, similar to the browser-based

approach, is preventing access to some useful entertainment and social media websites. Besides that, children may easily bypass this approach by accessing the World Wide Web through a different search engine.

c) *Monitoring Software*. It tracks children's activities on the Internet to allow parents to review records of these activities. This approach records activities without preventing objectionable content; therefore, children are at risk when browsing the Internet. Integrating monitoring software to provide an extra layer to filter the objectionable contents and block potentially harmful things such as location tracking, calls, and services that contain viruses and malware is possible. This approach provides advanced filtering compared to web browsers and search engines. Although bypassing this approach is more complicated than the previous approaches, children are able to bypass it with the help of a proxy website. Monitoring software may also include time monitoring and control to limit the time a child can access the Internet. Parents use time control software to enforce time limits to prevent children from using the Internet, for example, when parents are not present or late at night. Aside from the main adopted approach, this approach is usually included as a feature of the cyber parental control software and tools.

d) *Filtering Framework*. It aims to control the displayed contents for children to ensure their safety on the Internet by adding a defence layer to prevent objectionable content. Advanced developments in statistics have provided us with many new and enhanced means for mining web content by using, for example, web mining methods and techniques. Web mining is discovering and extracting information and knowledge from website documents using data mining methods and techniques (Etzioni, 1996). The literature decomposes web mining into three categories: content mining, structure mining, and usage mining (Anami et al., 2014; Raymond Kosala & Hendrik Blockeel, 2000).

Cyber parental control literature and tools use web content, rather than structural or usage mining, to detect and categorise contents on the cyber network. Several methods and techniques are used in the literature for web content mining based on the type of content, including text, hypertext, image, video, or audio. Literature categorises these contents into two groups; textual and visual. Examples of methods used for mining textual web content are Support Vector Machine (SVM), Neural Network (NN) (Chau & Chen, 2008), keyword-based (Dilip Patel & Pandya, 2017), blacklisting and whitelisting (Ahmadi et al., 2011), and filtering by statistical classification (Caulkins et al., 2006).

This study focuses on a filtering software approach for its advantages among other approaches, which are:

a) *Specificity*. Software filtering (SOF) allows applying advanced web mining techniques on the content of the webpage, which supports the goal of providing an SOF (Chiang et al., 2015).
b) *Adaptability*. With the internet's dynamicity, software filtering analyses each page's actual content in real-time rather than relying on predefined lists or categories of subjects (Ali et al., 2017).
c) *User Customisation*. Although other approaches might include this characteristic, the filtering software is tailored to account for the user's age, interests, and maturity level by customising the filtering subjects, which aligns with the aim of this study (Nagulendra & Vassileva, 2016).

2.1.3. Filtering Methods

Cyber filtering determines whether to block or allow some contents and connections of the cyber network based on predefined rules. Previous studies adopted an automated filtering approach, which combines web mining and content filtering techniques with several approaches. Examples of such approaches are IP Packet-based, URL and DNS-

based, keyword-based, and content-based approaches (Chapman, 1992; Moore, 2019; Nanda et al., 2008).

a) *PICS-based*. World Wide Web Consortium (W3C) created PICS, which uses metadata to determine the scope of the web pages and label websites. PICS aims to control users' access to the Internet, such as children and students (P. Y. Lee et al., 2003). This method relies on the metadata of the webpage, which some firms mislabel deliberately, and therefore, using PICS only is unreliable. In 2007, W3C proposed a Protocol for Web Description Resources (POWDER) to overcome the drawbacks of the PICS method (W3C Recommendation, 2009, 1 September). POWDER specifies a protocol for publishing metadata to enhance the filtering of web pages, and yet, handling the mislabeling of metadata is still a challenge for POWDER.

b) *IP Packet-based Filtering*. Routers and other network equipment use IP packets to pass data through the cyber network. IP packet filtering method filters these data packets based on their headers. There are two types of IP packet filtering: layer three and layer four (Varadharajan, 2010). These two types differ in their granularity and resource consumption on the filtering system. Literature uses two ways to implement IP packet filtering. The first way is through deploying firewalls on all connections. The second way is through employing existing network routing protocols to forward traffic for the relevant addresses to a "black hole" that discards the packets. The IP packet filtering causes two main collateral damages (Varadharajan, 2010). Firstly, it may lead to over-block unobjectionable websites and domains that use the same IP address as the truly objectionable websites. Secondly, it is easy to evade this filtering method by creating a different IP address attached to the same server for objectionable content.

c) *URL and DNS-based Filtering*. This method filters web pages based on comparing the URL or the DNS of the requested site against two reference lists, a whitelist and

a blacklist. The whitelist contains the allowed URLs or DNSs, while the blacklist contains the blocked URLs. Being lightweight is the advantage of this method. The efficiency of this method relies on reference lists, which mostly fail due to the incompleteness of these references because of the dramatic growth of the websites every day (users add 100,000 websites every day (Netcraft, 2018)). Examples of the most recent work on the URL-based filtering method are as Table 2.1 shows.

Table 2.1: The related studies that adopted the URL-based filtering method

Reference	Characteristic	Strength	Limitation
(Rajalakshmi et al., 2020)	The study proposes a URL classifier specifically designed for kids using a Recurrent Convolutional Neural Network (RCNN).	The classifier is designed to filter out inappropriate content for kids, providing a safer online environment.	The classifier may have false positives and negatives, potentially blocking appropriate content or allowing inappropriate content.
(Rao et al., 2020)	The study presents CatchPhish, a system for detecting phishing websites by inspecting URLs.	The system is able to detect phishing websites with high accuracy, which can help protect users from online scams.	The system is unable to detect new or sophisticated phishing techniques that do not rely on URL manipulation.
(Sahingoz et al., 2019)	The study discusses a machine learning approach to detect phishing from URLs.	The approach effectively detects phishing URLs, which can help prevent cyber attacks.	The study does not discuss how the approach can be scaled or how it performs in real-world applications.

(Kaptur & Kniaziev, 2019)	The study presents an adaptive method for complex internet content filtering.	The adaptive nature of the method allows it to adjust to changing internet content and maintain effectiveness.	The study does not provide a detailed analysis of the method's performance in different scenarios.
(Hussain et al., 2018)	Uses ontology-based approach for multilingual URL filtering	High accuracy in filtering multilingual URLs	Limited to the comprehensiveness of the ontology used
(Zhao et al., 2018)	The study proposes a stacking approach to identify objectionable-related domain names by analyzing passive DNS traffic.	The approach can effectively identify objectionable-related domain names, which can help block inappropriate content.	The study does not discuss the false positive rate of the proposed approach.
(Ali et al., 2017)	The study presents a web content classification system based on fuzzy ontology and Support Vector Machine (SVM).	The system can handle uncertainty and vagueness in web content and accurately classify it into different categories.	The study does not discuss the scalability of the system.
(Andriansyah et al., 2017)	The study discusses the development of an Indonesian corpus of pornography using a simple NLP-text mining approach.	The corpus can be used to support the Indonesian government's anti-pornography program.	The study does not discuss how the corpus was validated or how effective it is in real-world applications.

(Feroz & Mengel, 2015)	Uses URL ranking for phishing URL detection	High accuracy in detecting phishing URLs	Limited to the effectiveness of the URL ranking technique
(L. H. Lee et al., 2015)	The study proposes a method to filter objectionable content by mining browsing behaviours.	Browsing behaviours provide a personalized and dynamic way to filter content.	The variability of individual browsing behaviours can influence the method's effectiveness.
(Kotenko et al., 2014)	The study analyses and evaluates various webpage classification techniques for blocking inappropriate content.	The study provides a comprehensive analysis of different techniques, which can be helpful for researchers and practitioners in the field.	The study does not propose a new technique but evaluates existing ones.

d) *Keywords-based Filtering*. This method blocks web pages based on a comparison of some selected keywords with the contents of the webpage. Being lightweight is an advantage of keyword-based filtering. Its efficiency relies entirely on the selected set of keywords, which fails to consider the context of the keywords. For instance, if a webpage contains the word "sex" to refer to gender, this approach would deny the web page's access (B. K. Narayanan et al., 2018). Using Intelligent Content Analysis (ICA) overcomes the shortcomings of this method by considering the context of the keywords on the webpage. The disadvantage of using ICA, however, is the latency resulting from the complexity of the semantics' computation (Lee et al., 2003). Examples of relevant works on the keyword-based filtering approach are as Table 2.2 shows.

Table 2.2: The related studies that adopted the keyword-based filtering approach

Reference	Characteristic	Strength	Limitation
(Narwal, 2020)	Filters unethical and harmful content from web pages, using the cosine measure of similarity for comparing the webpage block content, alternate image text, and image tooltip with a dictionary of objectionable words	The system doesn't block the entire website but filters out the unethical blocks from the webpage, providing a clean webpage for kids. It uses a combination of keyword filtering and content analysis.	The system is unable to filter out all harmful content if it's not included in the dictionary.
(Altay et al., 2019)	Use a combination of three supervised machine learning techniques: SVM, maximum entropy (MaxEnt), and extreme learning machine (ELM).	High detection accuracy of 98.24%, and use of a large dataset of one hundred thousand web pages for evaluation	Time-consuming data preparation phase and the assumption that pages from Alexa are benign
(Zeniarja et al., 2018)	A search engine designed specifically for children, using a Naive Bayes Classifier to filter and rank documents based on their safety for children.	It ensures the safety of children by effectively filtering out unsafe websites.	The search engine lacks relevance of the query with the document being generated due to the lack of weighting techniques and algorithms to measure the

			similarity of the document.
(Dilip Patel & Pandya, 2017)	Categorize articles on child development and parenting contexts based on age categories.	The model can handle large data and does not require manual cataloguing. It can accurately categorize articles based on the context and content.	The model struggles with complex web pages and is unable to always find keywords related to the text.
(Kotenko et al., 2014)	It considers the analysis of text, HTML tags, and URL addresses to automatically categorize and filter web pages.	It supports the classification of different languages.	The boundary between the categories is often subjective, leading to problems when training the classifier.

e) *Content-based Filtering*. This method filters web pages based on their content. Previous studies used content-based classifications and filtering in many applications such as information retrieval, organising web-based information sources, search queries, and web pages (Kumbhar, 2012). Previous studies on web classification started a long time ago and started by addressing web mining in general. A few years later, some studies focused on web classification and filtering specifically by addressing web filtering issues and challenges. Examples of the relevant works on the content-based filtering method are as Table 2.3 shows.

Table 2.3: The related studies that adopted the content-based filtering approach

Reference	Characteristic	Strength	Weakness
(Shyry & Jinila, 2021)	Spam detection and prevention system using Collaborative and Content-based filtering, analyzed through UCI corpus experiments	Demonstrates superior performance of Content-based filter over Collaborative filter	Does not delve into other crucial factors like computational efficiency, scalability, and false positive/negative rates
(Modi & Jagtap, 2018)	Uses NLP and WSD, which significantly improved the accuracy of site classification	The method handles different types of web content, including text, images, and videos.	The performance of their proposal depends on the quality and coverage of the keyword database, which does not give flexibility for classifying diverse web pages.
(Ali et al., 2017)	Combines SVM and fuzzy ontology for website classification to reduce misrecognizing medical content as adult content	It can classify web pages into multiple categories (adult, medical, normal) instead of binary classification (adult vs non-adult)	Use a blacklist for classification, which may not include all potential adult content sources.
(Narwal & Sharma, 2016)	Classifying and filtering web content using webpage segmentation, feature extraction, and machine learning algorithms	Ability to distinguish between main content and noise on a webpage	Explored on small and similar web pages, and the model's performance may diminish when the feature set size increases beyond 20 features

(J. H. Lee, Yeh, & Chuang, 2015)	It combines Simplified Swarm Optimization (SSO), Genetic Algorithm (GA), Bayesian Classifier, and K-Nearest Neighbor (KNN)	It considers both HTML tags and terms simultaneously, improving classification accuracy.	Using SSO, despite its high accuracy, requires more computation time compared to other algorithms.
(Duan & Zeng, 2013)	It applies the topic modeling technique to establish a semantic model for objectionable content.	The approach offers superior detection and false alarm rates compared to traditional keyword-based methods.	It is limited to Chinese sentences and needs refining to incorporate essential stop words and short phrases with clear semantic description ability.
(Zeng et al., 2013)	It detects objectionable web text content by incorporating semantic analysis through the use of a topic modeling technique	Enableing fine-grained sentence-level detection of objectionable text	Based on a Chinese sentence dataset and heavily depends on the suitability of the dataset

This study creates a comprehensive subject-oriented approach by merging both URL-based and content-based as a two-tier filtering approach, using topic modeling, for the following advantages:

a) *Enhanced Accuracy*. Combining the two can yield better accuracy than either method used independently. While URL-based filtering can quickly block or allow known URLs, content-based filtering helps to analyse and classify new or unknown websites based on their content (Wu & Hwang, 2013).

b) *Context Awareness*. Content-based filtering can offer context to URL-based filtering, minimising the potential for false positives (blocking safe content) or false negatives (allowing harmful content) (Ibrahim et al., 2018).

c) *A balance between Speed and Depth*. URL-based filtering is faster but shallow; content-based filtering is slower but offers a deep understanding of web content. The combination ensures a balance between speed and depth of filtering (Ali et al., 2017).

2.2. Web Mining

Web mining is discovering and extracting information and knowledge from website documents using data mining methods and techniques (Etzioni, 1996). It is an integrated field involving a few research areas, such as informatics, statistics, data mining, and computational linguistics (Jicheng et al., 1999). Although data mining research started more than two decades ago, it became more significant because of the dramatic growth of the availability of information resources on the web (Berardi et al., 2015; Raymond Kosala & Hendrik Blockeel, 2000). The web mining process includes four phases as follows:

a) *Resource finding*. This initial phase of the web mining process gathers specific documents and data resources from a selected website.

b) *Pre-processing and information selection*. This is a crucial phase to cleanse, transform, and standardize the data and remove any irrelevant information from the collected websites to make it suitable for further analysis (Dwivedi & Rawat, 2015).

c) *Generalisation*. This phase discovers and finds general patterns on the website automatically. The goal is to extract high-level knowledge from low-level data.

d) *Analysis*. The final phase of the web mining process involves validating or interpreting the discovered patterns from the websites. It includes applying statistical tests to confirm the significance of the patterns, using visualization tools to better

understand the patterns, or incorporating domain knowledge to interpret the implications of the patterns. Topic modeling and content classification are examples of such analysis (Porouhan & Premchaiswadi, 2017). The main aim is to provide actionable insights that can be used for decision-making, such as filtering objectionable web content

The state-of-the-art decomposes web mining into three categories: web content mining, web structure mining, and web usage mining, as Figure 2.2 describes (Anami et al., 2014). The following points highlight web structure and usage mining, while the following section addresses web content mining in detail.

a) *Web Structure Mining*. It focuses on uncovering the model that lies beneath the link structures of websites, a task accomplished through hyperlinks to identify graph patterns (Ehikioya & Zeng, 2021; Tyagi & Gupta, 2018). Its significance lies in its ability to enhance the quality of indexing for search engines, making it an essential tool in the era of digital information. Techniques such as association rules and clustering are often utilized in conjunction with popular algorithms, including HITS (Kleinberg, 1999), Page Rank (Page et al., 1999), Weighted Page Rank (Xing & Ghorbani, 2004), and Eigen Rumor (Fujimura et al., 2005). These methods of web structure mining have been integrated into several tools, showcasing the wide range of applications for this technology. However, it is unsuitable for web content classification and filtering.
b) *Web Usage Mining*. It extracts user patterns from weblog records to identify user behavioural models, playing a vital role in business-focused websites aiming to enhance customer satisfaction (Anami et al., 2014; Ehikioya & Zeng, 2021). Among the variety of techniques employed, the association rule is the most utilized, enabling developers to construct and adjust web pages more effectively based on the presence

or absence of these rules. However, it is unsuitable for web content classification and filtering.

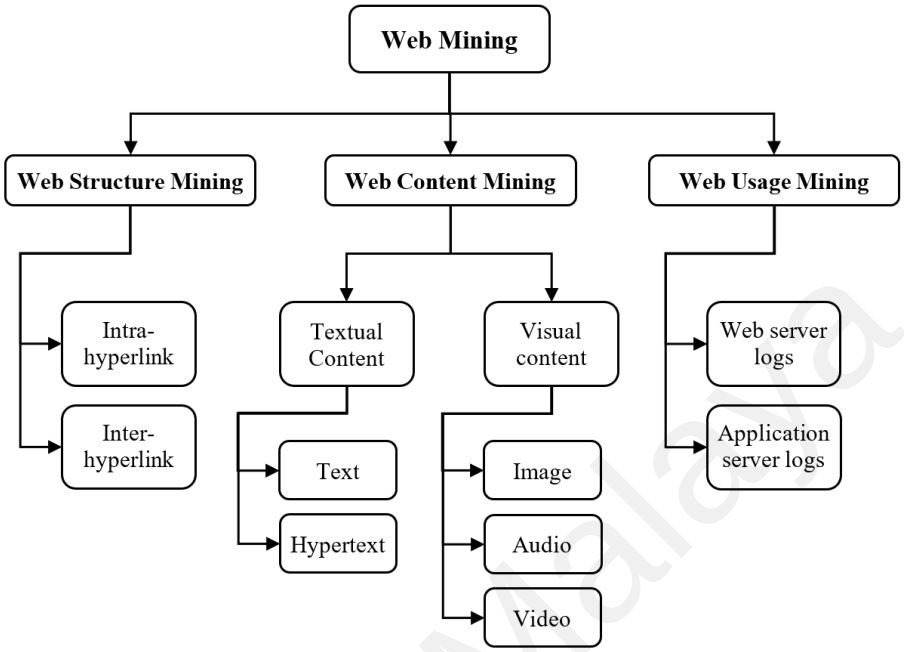

Figure 2.2: The taxonomy of web mining

2.3. Web Content Mining

Web content mining focuses on the contents of a website itself. Websites contain several data types; text, hypertext, image, video, and audio. These data can be categorised into two groups: textual and visual. Web content mining aims to help find data and filter it for the user, so it is usually performed based on the preference or demand of the user. Techniques and algorithms are essential to perform web content mining. The following subsections address the web content mining techniques and algorithms' characteristics, applications, limitations, and strengths. It then addresses the evaluation metrics of web content classification used by the state-of-the-art.

2.3.1. Web Classification Technique

The majority of web pages contain textual and image contents, which together constitute a topic or more inside the webpage. Text and image contents describe a website efficiently, which are essential sources for classifying web pages effectively (S. Liu &

Forss, 2015b). Literature uses three techniques for web content classification which are textual, visual, and topical. The following subsections address each of these approaches.

a) *Textual Technique*. The textual technique in web structure mining involves the use of text classification algorithms to categorize the web, a crucial process aimed at assigning structured documents to specific categories (Chau & Chen, 2008).
b) *Visual Technique*. The visual technique in web structure mining predominantly relies on image classification, which seeks to categorize images based on their contextual information. This approach has been a focus of research for the past two decades and encompasses three techniques: (a) keyword-based, (b) blacklist-based, and (c) content-based classification and filtering.
c) *Topical Technique*. The topical technique in web structure mining primarily relies on topic modeling algorithms, which identify abstract topics or patterns across a dataset for effective classification, clustering, sorting, and predicting a large corpus of documents.

The scope of this study falls within using textual content of web data, and therefore, this section focuses further details on textual and topical techniques. Adapting the textual techniques on web classification and filtering faces several distinct challenges, resulting in drawbacks, including:

a) Its design originally catered to structured data, whereas web content comprises different types of structured, semi-structured, and unstructured data.
b) Struggling with handling the diverse types of data found in web content and the added intricacies of managing hypertext connections.

On the other hand, the advantage of adapting the topical technique including:

a) *Effectiveness*. Detecting and filtering objectionable web content allows for practical applications in fields like software engineering and topic evaluation.
b) *Wide Utility*. Topic modeling has been utilized in webpage and website classification, clustering, detecting spam, discovering trending topics, and identifying prominent subjects in Q&A and review websites.
c) *Scalability*. Given the fluid nature of the internet and the increase of content topics, topic modeling has the ability to interpret new and changing topics on the web.
d) *Customization*. Using topic modeling, the classification and filtering of the web content are adjustable to factor in the user's age and culture by customizing the filtering topics.

Despite the broad applications and benefits of topical techniques, a major limitation highlighted in recent studies is the neglect of the unique structures of web content data. This oversight leads to missed topics and lowers the topic quality. Some studies have addressed this issue, proposing models like the Named Entity Topic Model (NETM) and a topic-graph probabilistic personalization model for web search, but even these models have failed to consider the HTML structure of webpage content (Altarturi et al., 2023). Table 2.4 tabulates a few recent related studies that utilize topic modeling on the web, their technique, application, and some limitations.

Table 2.4: The related studies that utilized topic modeling in web applications

Study	Technique	Application	Limitations
(Lee & Cho, 2021)	LDA and word2vec	Webpage classification and ranking	Train on small datasets and neglect the HTML structure of the webpage content.
(Zhao et al., 2021)	Topic-graph probabilistic personalization	Web search personalization	Neglects the webpage's structure and assumes that a clicked webpage includes interesting

	model using the LDA		topics and a skipped webpage covers non-interesting topics.
(Asdaghi et al., 2020; Wan et al., 2015)	LDA with content and URL-based analysis	Web spam detection for search engine	Only utilizes the LDA topic model and is not generalised to different types of web content.
(Yang et al., 2020)	Named Entity Topic Model (NETM)	Web content popularity growth for news articles	Lack of interactions between named entities and semantic topics due to neglecting the web content
(Sayadi et al., 2015)	Semi-supervised LDA with Random Forest	Multilayer soft web classification	Ineffective for all types of web pages or text content due to neglecting the structure of the web.
(Alghamdi & Selamat, 2015)	LSA and pLSA	Webpage clustering for Arabic	Relying only on pLSA results in low coherent topics, thus, low classification accuracy when testing on a variety of web pages
(Liu & Forss, 2015a, 2015c)	LDA, along with SVM	Classification of harmful web pages	The performance in detecting violent content was disappointing, and the effect of relabeling efforts was limited.
(Chen & Zhou, 2014)	Modified LDA, along with the KNN classifier	Web clustering	The clustering performance is not significantly improved with the increase in the number of users, and the performance goes down when the number of topics increases.

2.3.2. Web Classification Algorithm

The state-of-the-art commonly used several algorithms for web classifications due to their outstanding performance. The following subsections address these algorithms as follows:

a) *Support Vector Machine (SVM)*. It is a highly effective solution for classification problems proposed by Cortes and Vapnik in 1995 (Cortes & Vapnik, 1995).

b) *Naïve Bayes*. It is a straightforward yet effective algorithm based on Bayes' theorem, with a strong assumption of conditional independence among attributes given the class.

c) *Random Forest (RF)*. The Random Forest algorithm, introduced by (Breiman, 2001), is a highly effective general-purpose classification and regression technique, especially potent in scenarios with a large number of variables and observations. Combining numerous randomized decision trees and averaging their predictions demonstrates exceptional performance.

d) *Logistic Regression (LR)*. Introduced by (Berkson, 1944) and further evolved by (Cramer, 2002), it is a powerful method for predicting a categorical outcome variable from one or more categorical or continuous predictor variables. By modeling the probability of a particular outcome based on individual attributes, LR calculates odds ratios in the presence of several explanatory variables.

e) *K-Nearest Neighbor*. The KNN algorithm, a prevalent classification technique introduced by Cover & Hart (1967), uses prototype examples and a training set of pattern vectors from each class for classification. An unknown vector is classified based on the majority rule from its 'k' nearest prototype neighbours, ideally with 'k' being an odd number to avoid ties and overlap zones.

Each of these algorithms has its advantages and disadvantages when applied to web content classification. The advantages of SVM in web classification are:

a) Its primary strengths lie in its robust and high-dimension learning technique.
b) Its ability to converge to the global minimum of the specified error function.
c) It results in classifying high-dimensional web content data.

However, SVM is theoretically complex (conceptual foundations and mathematical formulations are intricate) and computationally expensive, which results in long training times and sensitivity to noisy data and outliers.

The advantages of NB in web classification are:

a) Despite often violating this assumption in real-world applications,
b) It often delivers competitive classification accuracy, making it a popular choice for classifying web content data (Asdaghi & Soleimani, 2019).
c) Its computational efficiency, practicality, and ability to estimate the posterior probability of each class given an object further bolster its usage.

However, its oversimplification due to the assumption of independence can sometimes lead to poorer performance (Kelly & Johnson, 2021), particularly with interdependent features commonly found in web content data.

The advantages of RF in web classification are:

a) Flexible handling of large-scale web pages.
b) Capable of providing variable importance metrics for the webpage (Krishnani et al., 2019; Kumar et al., 2018).

However, its potential limitations include possible overfitting with noisy data, computational intensity with very large datasets, and less interpretability due to being trained on different but overlapping subsets of the dataset (Ihekoronye et al., 2022; Priyadharshini et al., 2023).

The advantages of LR in web classification are:

a) Ability to provide probabilistic interpretations and handle categorical and continuous inputs.
b) It is highly interpretable compared to many other techniques (Martin et al., 2021).

However, its limitations include the need for large sample size for stable results, its assumption of linearity of independent variables and log odds, and difficulty in handling complex, non-linear relationships and high-dimensional data, which are often characteristics of web content data (Manotas & Gonzalez-Perez, 2020).

Despite the simplicity of the KNN algorithm, the algorithm often achieves low error rates in practice in web classification (Kumari & Soni, 2017). However, it has several disadvantages, including:

a) Its computational complexity is due to a large number of distance computations, which is particularly problematic when dealing with vast and high-dimensional web content data (Bijalwan et al., 2014).
b) It struggles with imbalanced data and is sensitive to the choice of 'k' and the distance metric used (Leguen-deVarona et al., 2020).

2.3.3. Web Classification Evaluation

The evaluation of web content classification relies on four calculations; True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN), as Table 2.5 illustrates. Four metrics utilize these calculations to measure the predictivity performance of a web content classification, which are:

a) *Accuracy*. It is defined as the number of correct predictions divided by the total number of predictions. The accuracy computation is shown in the following equation:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

b) *Precision*. The precision is defined as the number of TP over the number of true positives plus the number of FP is shown in the following equation:

$$Precision = \frac{TP}{TP + FP}$$

c) *Recall*. The recall is defined as the number of TP over the number of TP plus the number of FN given in the following equation:

$$Recall = \frac{TP}{TP + FN}$$

d) *F1-Measure*. The f1-measure is defined as the harmonic mean of precision and recall, as shown in the following equation:

$$F1 = 2\frac{Precision \times Recall}{Precision + Recall}$$

Table 2.5: Confusion matrix of evaluating web content classification

		Predicted	
		Positive	**Negative**
Actual	**Positive**	TP	FN
	Negative	FP	TN

2.4. Existing Frameworks

This section states a comparison of the most related existing frameworks that are applied as cyber parental control. This comparison is based on their filtering technique, classification method, datasets, and evaluation metrics, as Table 2.6 summarizes.

Table 2.6: Comparison of the existing frameworks

Reference	Filtering technique	Classification method	Dataset	Evaluation
(Narwal, 2020)	Keyword-based & content-based	Textual content mining approach using ANN	140 websites No objectionable category Not available publicly	Accuracy, precision, recall, & F-measure
(Rajalakshmi et al., 2020)	URL-based	URL features using CNN	92,560 URLs, kids' category & not available publicly	Accuracy
(Rao et al., 2020)	URL-based	URL features using TF-IDF	126,077 websites No objectionable category & not available publicly	Accuracy, precision, & F-measure
(Altay et al., 2019)	Keyword-based & content-based	Textual content mining approach using SVM	228,848 URLs but no objectionable category, available publicly	Accuracy
(Sahingoz et al., 2019)	URL-based	Textual content mining approach using NLP	73,575 URLs No objectionable category & not available publicly	Accuracy
(Hussain et al., 2018)	URL-based & keyword-based	Textual content mining approach using an ontological approach	65,000 URLs of Blacklisting and whitelisting categories & are not available publicly	Accuracy, precision, & F-measure
(Zhao et al., 2018)	URL-based	URL features using CNN	11,121 objectionable websites & not available publicly	Accuracy

(Liu & Forss, 2015b)	Content-based	Topical content mining using LDA	80,000 URLs No objectionable category Not available publicly	Precision & recall
(Patel et al., 2015)	Keyword-based	Textual content mining approach using quantum-based NN	2,000 objectionable URLs & not available publicly	Accuracy
(Kotenko et al., 2014)	URL-based & content-based	URL features and textual content mining using TF-IDF	No objectionable category & not available publicly	Accuracy, recall, f-measure, & precision
(Duan & Zeng, 2013)	Content-based	Topical method using LDA	4,290 objectionable Chinese sentences & available publicly	TP and FP
(Zeng et al., 2013)	Content-based	Topical method using LDA	35,500 objectionable Chinese documents & available publicly	TP and FP

While many studies have made important strides in the field of cyber parental control, a critical analysis of these studies represented in Table 2.6 reveals significant gaps in the state-of-the-art of this field, including:

a) Lack of a comprehensive and publicly available ground truth dataset. The ground truth dataset is crucial for developing and testing new approaches in the context of cyber parental control.
b) Lack of adapting topical techniques to classify and filter web content. Neglecting the topical techniques results in a limitation in covering a wide range of topics and flexibility for customization of the developed framework.

c) Overlooking the potential of advanced topic modeling in enhancing the effectiveness and accuracy of the developed framework to classify and filter objectionable web content.

Topic modeling provides an advanced approach to classifying and filtering objectionable content. This study aims to fill these gaps by exploring the utilization of topic modeling within the cyber parental control framework and enhancing its coherency to improve the effectiveness and accuracy of classifying and filtering objectionable web content.

2.5. Summary

This chapter underlines the current state-of-the-art web classification and filtering techniques and methods used in cyber parental control. It starts by introducing the cyber parental control term and its related concepts. It discusses the techniques for web content categorisation used in this study and the approaches to online parental control, including filtering frameworks. Central to this research, state-of-the-art filtering methods are then reviewed, including the web mining techniques and filtering methods employed in this study.

Classifying web content by its topics requires an essential understanding of the combination of content filtering and web content mining techniques. While the existing studies demonstrated significant progress and achievements, the review in this chapter shows that there is still a gap in considering the potential of advanced machine learning algorithms in enhancing web content filtering in cyber parental control. This research aims to fill this gap by utilizing topic modeling to enhance the effectiveness of classifying and filtering objectionable contents in the cyber parental control framework. The following chapter discusses the fundamentals of topic models and their utilizations in web applications.

CHAPTER 3: TOPIC MODELING MATHEMATICAL BACKGROUND

An immense volume of hypertext and digital documents exist online and offline with content that can offer useful information and insights. Topic modeling, a textual web content mining technique, aims to discover latent semantic structures or topics within a set of textual digital documents. Topic models are widely applied in the generative language models, spam filtering, summarisation, sentiment analysis, text categorisation, text similarity, content classification, and recommender system (Chung et al., 2019; Guo et al., 2018; Linton et al., 2017; Liu et al., 2018; Mulunda et al., 2018). Web applications use topic modeling to categorize, classify, index, and improve search and recommendation systems by understanding the latent semantics and topics of the webpage, as Figure 3.1 illustrates.

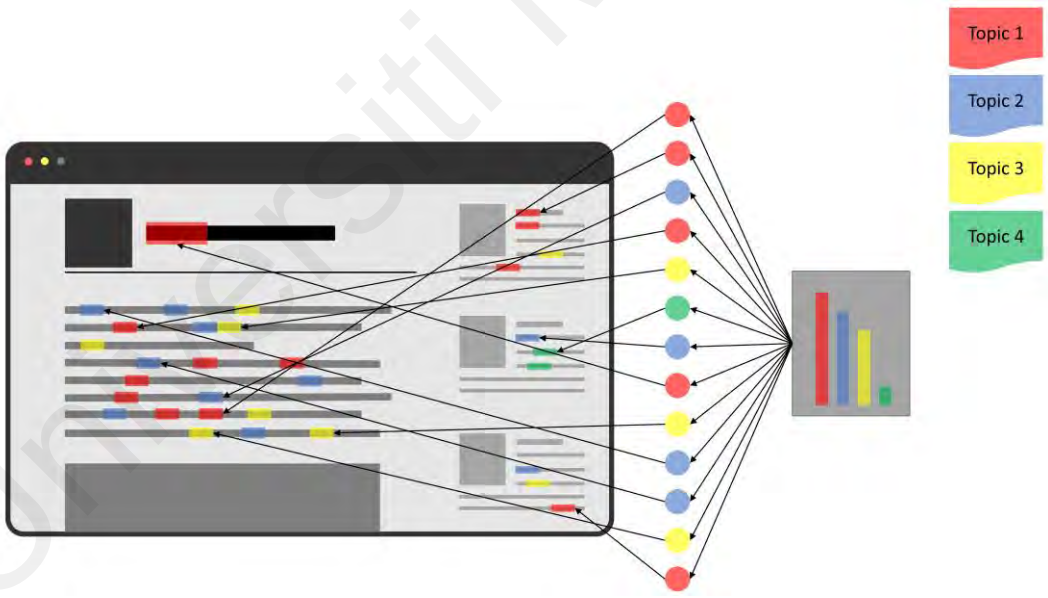

Figure 3.1: Understanding the latent semantics and topics of the webpage

Despite their design differences and applications, topic models are based on distributional and statistical models. This chapter provides an overview of the mathematical background of topic modeling, and then it synthesises the state-of-the-art topic models into a taxonomy highlighting each category and its topic models. This

chapter also presents in brief non-probabilistic topic modeling. The benchmark models used in this research are then illustrated, and the tools to apply and develop topic modeling are highlighted.

3.1. Mathematical Background

This section presents the mathematical background for topic models in general. It categorizes the related models commonly used in topic modeling, as Figure 3.2 illustrates.

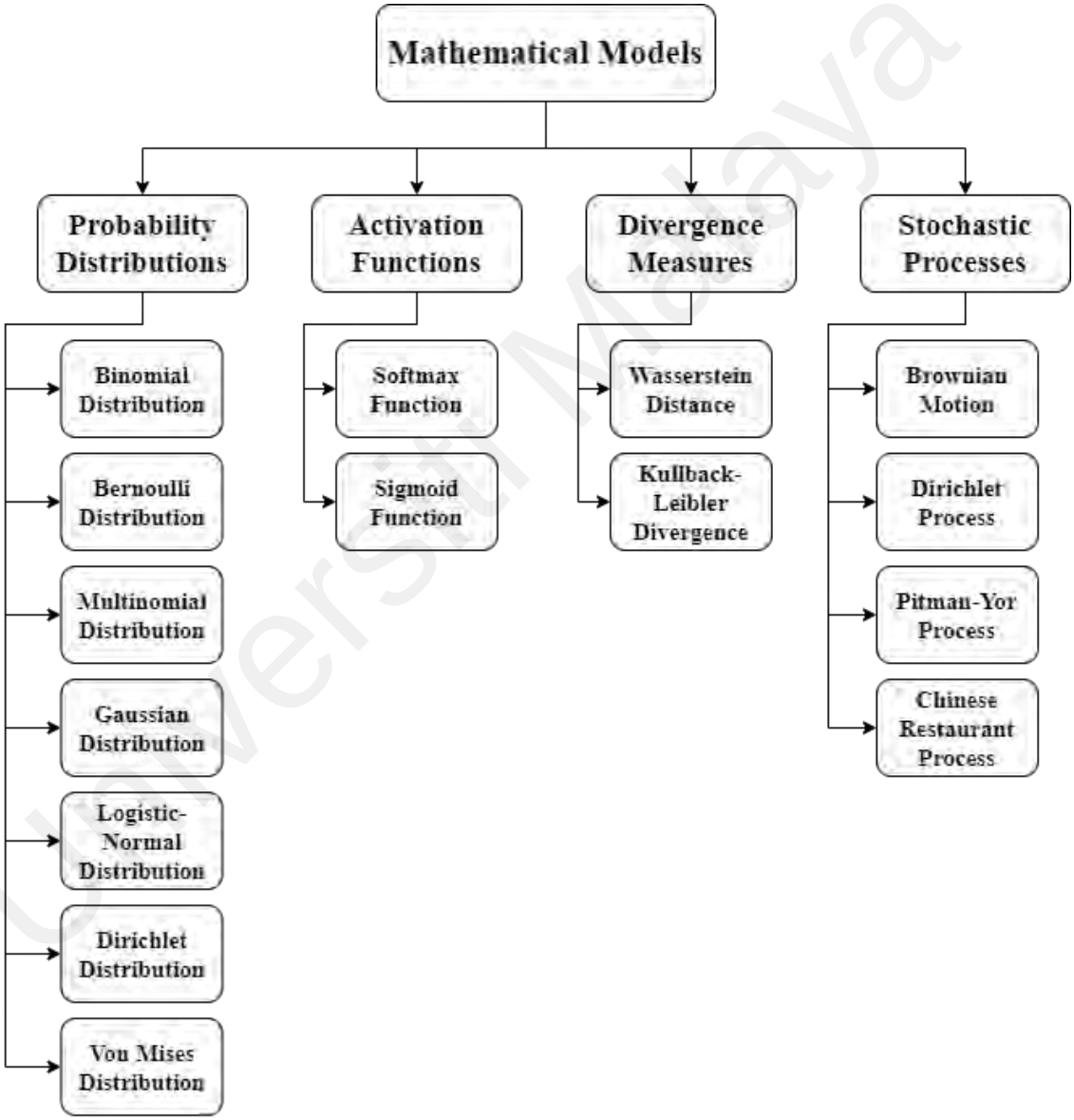

Figure 3.2: Taxonomy of topic models' mathematical background

The description of these categories is as follows:

a) *Probability distributions*. Probability distribution models in topic modeling, known as probabilistic models, offer interpretability and flexibility, accommodating various data types and assumptions, and they allow for the incorporation of prior knowledge and quantification of uncertainty (Roberts, 2021). However, their performance is heavily dependent on the validity of their assumptions, and they can be computationally intensive, especially with large datasets (Ruan & Stormo, 2017). Choosing the right model for a specific task can be challenging due to the variety of models available, and there's a risk of overfitting, particularly with complex models or small datasets. Examples of such probabilistic models used in topic modeling are Bernoulli, multinomial, gaussian, logistic-normal, Dirichlet, and Von Mises Distributions.

b) *Activation functions*. Topic modeling utilizes activation functions, mainly Softmax and Sigmoid. Softmax transforms the raw outputs of a classier into probabilities, whereas Sigmoid maps inputs to a value between 0 and 1 for binary classification. These functions allow for a probabilistic interpretation of topics and words, assisting in the comprehension of model predictions, and their differentiability facilitates gradient-based optimisation techniques. However, they can result in "exploding" or "vanishing" gradient issues, resulting in unstable training. Therefore, their application to web content analysis should be carefully chosen based on the specific problem and data type (Doan & Hoang, 2021).

c) *Divergence measures*. Divergence measures, such as the Wasserstein distance and Kullback-Leibler (KL) divergence, provide a quantifiable measure of the difference between probability distributions of topics in documents and words in topics. While the Wasserstein distance considers the geometric structure of the data, making it less sensitive to small changes in the data distribution, the KL divergence provides a

measure of the information lost when one distribution is used to approximate another. The weakness of the Wasserstein distance is computationally intensive with high-dimensional data (Dai et al., 2020), and the KL divergence's lack of symmetry (the divergence of one distribution from another isn't the same in reverse) (Lesniewska-Choquet et al., 2019).

d) *Stochastic processes*. Stochastic processes in topic modeling capture the distribution of topics across documents and words within topics, providing a mathematical framework for handling randomness in topic and word generation. They offer scalability and flexibility, allowing for potentially infinite topics and a principled way to estimate topic distribution (Phadia & Phadia, 2016). However, they can be computationally intensive, and their assumptions may not always hold true (Rama, 2016). Examples of stochastic processes used in topic modeling include the Brownian motion, Dirichlet, Pitman-Yor, and Chinese Restaurant processes.

3.2. Probabilistic Topic Model Taxonomy

The Latent Semantic Indexing (LSI) model was the first statistical model for grouping co-occurrence terms in documents. LSI modifies the original dataset so that documents and terms pertaining to the same topic can be mapped. LSI considers that documents and words have multiple interdependencies. Thus, complications may arise when a word has numerous meanings. The case of numerous meanings of words is common in the huge textual dataset. Hofmann proposed a Probabilistic Latent Semantic Indexing (pLSI), an offshoot of LSI, to improve topic modeling (Hofmann, 1999; Vayansky & Kumar, 2020). The LSI model associates a word with a topic, while the pLSI model associates words with topics based on likelihood. By allocating each word to a subject taken from a multinomial distribution over topics, the model is constructed in a more meaningful way.

Belei et al. (2003) proposed the LDA, which extended the pLSI by adding Dirichlet priors on topic distributions (Alkhodair et al., 2018; Hajjem & Latiri, 2017). The LDA is

a generative model capable of modeling topics for unseen textual documents and can calculate the proportion of one variable given the value of another. So, it can be called a probabilistic graphical model. Although the LDA originally is an unsupervised text model, several studies modified it to work on labelled data as a supervised model (Mcauliffe & Blei, 2007).

LDA considers textual documents as a mixture of topics, so it is known as a mixture model or admixture model due to the fact that their segments are themselves a mix of other segments (Heinrich, 2005). The LDA generative model permits the explanation of a set of observations by a group of unobserved variables (Griffiths & Steyvers, 2004) and uses a multinomial distribution in order to generate the new document. The prior distribution over a topic will be used to calculate the probabilities to select certain topics. A topic is a distribution of words. The sampling of words in each topic starts after the topic is sampled. Each text has a unique proportion of topics, and each topic has a unique probability of words. In the given document, the proportion of all topics equals one. The presumptions of this process are:

a) A document is a collection of words where the order of words is not taken into account.
b) Each document contains a variety of subjects, whereas, for K subjects, each topic occupies a proportional fraction of the document.
c) Each topic contains a variety of words.

The LDA's plate notation is illustrated in Figure 3.3, where α and β are Dirichlet distributions, and θ and φ are topic distribution and word distribution, respectively.

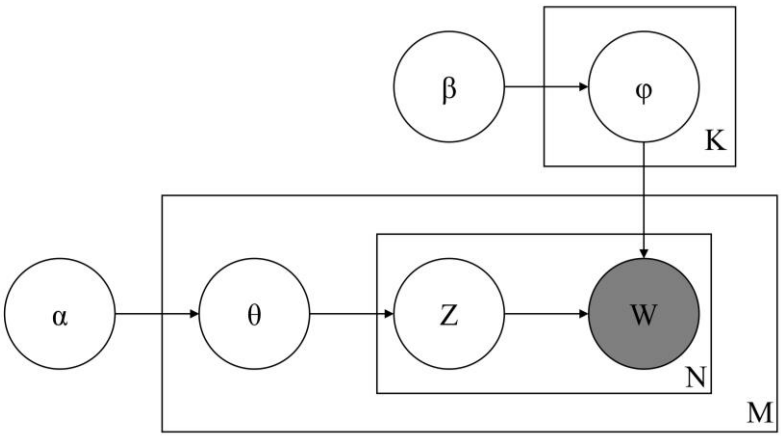

Figure 3.3: Latent Dirichlet Allocation plate notation

Several studies proposed topic models that add constraints on the traditional LDA to generate modified models called LDA variants. This section categorises these topic models based on their underlying statistical model, as the following subsections show. Figure 3.4 shows the taxonomy of these categories and their models.

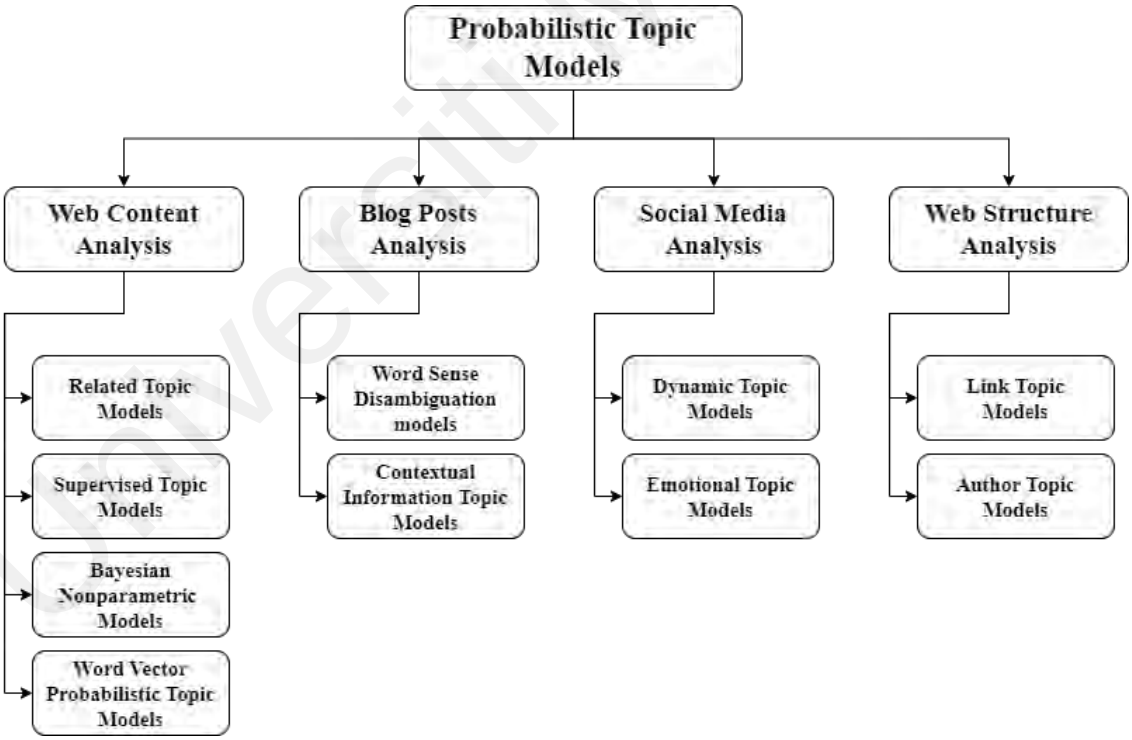

Figure 3.4: Taxonomy of the probability distribution topic models based on web application

3.2.1. Web Content Analysis

Web Content Analysis employs advanced topic modeling techniques to navigate the challenge of comprehending and classifying web content, transforming the seemingly

insurmountable task into an insightful process. Several topic modeling categories enable the extraction of meaningful insights from web content and, more importantly, facilitate the critical tasks of web content classification and filtering. The following categories serve as the linchpins in this endeavour:

a) *Related topic models*. These models capture the inherent correlation between topics in real textual data and corpus, providing a more nuanced and interconnected view of topics.
b) *Supervised topic models*. The models address the classification challenges posed by supervised learning in machine learning. They are characterized by their effective use of a document's ancillary information and ability to handle text classification difficulties more efficiently than other categories.
c) *Bayesian nonparametric models*. Characterized by their adaptability and flexibility, these models stand out for their ability to learn a number of topics from the data itself, eliminating the need for predefining this parameter, and their capacity to handle both semantic and syntactic hidden variables.
d) *Word vector probabilistic topic models*. These models focus on training word vectors, which significantly enhance the generalization performance of the topic model. They often result in learned keywords with a higher degree of semantic consistency, particularly effective when applied to learning short texts.

The strength of these topic models on web content analysis includes the following:

a) *Adaptability*. Models of this category adapt and learn from the data they encounter, including increasing the number of learning topics as the text corpus grows and enhancing the model's generalisation performance using trained word vectors.
b) *Handling Complex Structures*. Models such as Hierarchical Latent Dirichlet Allocation (HLDA), Syntactic Topic Model (STM), and Pachinko Allocation Model

(PAM) excel in processing topics hierarchically or describing the correlation between all topics, providing a comprehensive view of topic relationships.

c) *Topic Correlation*. Related Topic Models can describe the inherent correlation between topics, a feature often presented in real textual data and corpus.

d) *Enhanced Accuracy and Precision*. Supervised Topic Models effectively handle text classification difficulties, and using word vectors increases the categorization precision of the model. These models also improve the consistency and interpretability of topic words.

e) *Efficient Use of Information*. These models make efficient use of available information, whether making better use of a document's ancillary information, mining the semantic link between words more efficiently, or efficiently eliminating the subjectivity of artificially created labels.

f) *Semantic Consistency*. These models are particularly effective when applied to learning short texts, often resulting in learned keywords with a higher degree of semantic consistency.

However, applying these models includes several disadvantages, which are:

a) *Complexity*. Many of the topic models in the category are complex to implement and understand, such as the Hierarchical Dirichlet Process (HDP), Correlated Gaussian Topic Model (CGTM), PAM, and Hierarchical Pitman-Yor Process (HPYP), especially when the number of topics increases. This complexity is a significant challenge in scenarios involving large datasets or numerous topics.

b) *Posterior Inference*. Models like Bayesian nonparametric and supervised topic models often require numerical optimization algorithms. This challenge makes these models more difficult to interpret and increases the computational cost of using these models (Gupta et al., 2019).

The most common related topic models include Correlation Topic Model (CTM) (Lafferty & Blei, 2005) and PAM (Li & McCallum, 2006). Supervised topic models include Supervised LDA (sLDA) (J. Mcauliffe & Blei, 2007), Label-LDA (LLDA) (Ramage, Hall, Nallapati, & Manning, 2009), and Dirichlet-Multinomial Regression (DMR) (Guimaraes & Lindrooth, 2005). Examples of Bayesian nonparametric models are HDP (Teh, Jordan, Beal, & Blei, 2006), HLDA (T. Griffiths, Jordan, Tenenbaum, & Blei, 2003), and HPYP (Lim, Buntine, Chen, & Du, 2016). Finally, topic models utilizing the word vector probabilistic, including Pseudo-document-based Topic Model (PTM) (Zuo et al., 2016) and Gaussian LDA (GLDA) (Das, Zaheer, & Dyer, 2015).

3.2.2. Blog Post Analysis

Blog post analysis understands the content and context of web-based articles, and it primarily employs two topic modeling categories, which are:

a) *Word sense disambiguation*. WSD topic models determine the correct meaning of ambiguous words in web content. They use context to disambiguate word meanings, thereby improving the accuracy of text analysis by correctly interpreting word meanings.
b) *Contextual information*. These models are significant in analyzing web content where the context or the order of words is important. They are characterized by their ability to consider the surrounding context of words or phrases in the text.

These two categories have their strengths in analyzing blog posts, including:

a) Improving the accuracy of text analysis by correctly interpreting word meanings
b) Ability to capture nuances and subtleties that models only consider individual words might miss.
c) Handle ambiguity in the text by using the surrounding context to disambiguate the meaning of words or phrases.

However, applying these models includes several disadvantages, which are:

a) Struggle when the context does not provide clear clues for word sense disambiguation or when the order of words is important.
b) Complex and require large amounts of data to effectively capture context.
c) Complex to implement and understand and often requires a good understanding of NLP and machine learning techniques.
d) The effectiveness of these models can depend on the quality of the context and the context size. The models might not perform well if the context and the context size are not informative or misleading.

3.2.3. Social Media Analysis

A crucial aspect of understanding the digital landscape employs dynamic and emotional topic models to extract meaningful insights from the vast array of social media content, including the analysis of user reviews and comments. This analysis utilizes two main topic modeling categories as follows:

a) *Dynamic topic models*. These models are able to capture temporal properties and adeptly track the evolution of topics over time, providing valuable insights into trending topics and shifts in discourse. However, their complexity and the need for substantial data can pose challenges.
b) *Emotional topic models*. These models focus on sentiment analysis, detecting and quantifying the emotional content of the text to offer a window into public sentiment and mood trends. Despite their ability to provide insights into the emotional landscape of social media, these models can sometimes struggle with accurately detecting and quantifying emotions, particularly in shorter or more ambiguous texts.

Most common dynamic topic models, including DTM (D. M. Blei & J. D. Lafferty, 2006), the online LDA model (On-Line LDA) (AlSumait et al., 2008), and the Continuous

Dynamic Topic Model (cDTM) (Wang et al., 2012). Emotional topic models include Multi-Aspect Sentiment (MAS) (Zhu et al., 2009), Reverse Joint Sentiment Topic model (Reverse-JST) (Lin et al., 2011), Aspect and Sentiment Unification Model (ASUM) (Jo & Oh, 2011), and Supervised Joint Aspect and Sentiment Modeling (SJASM) (Hai et al., 2017) models.

3.2.4. Web Structure Analysis

Topic models of this category generally uncover hidden structures and relationships that might be overlooked by models that focus solely on text content. It includes two sub-categories as follows:

a) *Link Topic Models*. These models allow a better understanding of the interconnected nature of the web and are able to analyze web texts where hyperlinks play a pivotal role in revealing the latent structure of the text.
b) *Author Topic Models*. Authorship models are key to understanding the relationship between text topics and their authors. It is able to associate specific topics with particular authors, providing valuable insights into an author's interests and areas of expertise.

Two main advantages of these types of topic models, which are:

a) It also helps identify the relevance of a webpage based on the number and quality of its inbound links, or it can reveal clusters of related web pages based on their interlinking patterns.
b) It can help identify an author's recurring themes or preferred subjects, offering a unique perspective on the intersection of authors and their chosen topics.

However, these models also face several disadvantages, including:

a) The analysis's complexity and the need for data that includes link structure can pose challenges.
b) The model may face challenges when dealing with authors who write on a broad range of topics, as it could struggle to accurately associate such diverse topics with a single author.

The main common topic models that focus on link topic models are the Link-Latent Dirichlet Allocation (Link-LDA) (Cohn & Hofmann, 2000), Pairwise Link-Latent Dirichlet Allocation (Pairwise Link-LDA) (Nallapati, Ahmed, Xing, & Cohen, 2008), and the Relation Topic Model (RTM) (Zhang et al., 2013). The authorship topic models primarily utilize the Author Topic Model (ATM) (Steyvers et al., 2004) and its variants, such as the Author Conference Topic Model (ACT) (Tang et al., 2008).

Aside from these probability distribution topic models, early studies attempted to model document topics using non-probabilistic modeling, which the following section briefly addresses.

3.3. Non-Probabilistic Topic Models

Aside from the LDA-based probabilistic topic model, early studies attempted to model document topics using non-probabilistic modeling, starting by introducing the LSA (Kontostathis & Pottenger, 2006). The characteristics of these models are as follows:

a) Leveraging the matrix decomposition techniques like Singular Value Decomposition (SVD).
b) Disregard word order and represent the text corpus through the co-occurrence matrix of words and documents.
c) Extract the latent meaning of text from a collection of documents by analyzing the most frequent words in the documents.

Non-probabilistic topic models, while offering significant advantages in various domains, also present certain disadvantages that impact their effectiveness (O'callaghan et al., 2015), including:

a) These models have issues with polysemy (one word with numerous meanings) and synonymy (multiple words with the same meaning).
b) Non-probabilistic models have not found significant contributions in supervised fashions.
c) They face issues of factor overfitting, which is mitigated by retaining only the first k dimensions of the singular value decomposition matrix.

3.4. Topic Modeling Evaluation

Various evaluation measurements can be utilized to evaluate the performance of a machine learning model. Regarding topic modeling, literature uses perplexity measure, topic coherence measure, or both measurements. Several recent studies have argued that perplexity is less correlated to human interpretability and understandability (Li et al., 2016; Röder et al., 2015; Zuo et al., 2016) and does not address the goal of exploratory research of topic modeling. Thus, perplexity is no longer a general way of evaluating topic models (Zuo et al., 2016). The following subsection illustrates and describes each metric.

Topic coherence calculates and measures the consistency and quality of each individual topic with reference to the semantic similarity between the words in the topic or how many the words of each individual topic occur within the same set of documents (Aletras & Stevenson, 2013; Li et al., 2016; Mimno et al., 2011). The authors (Mimno et al., 2011) introduced the topic coherence metric, which produces a stronger correlation with human judgments in evaluating topic quality (Arora et al., 2013; Chen & Liu, 2014). Topic coherence indicates the quality of the model and how accurate the terms are. The

higher the topic coherence score indicates more topic coherence, the more efficient the model is. There are a few coherence calculations in the literature; C_V (Lau et al., 2014; Syed & Spruit, 2017), C_{UCI} (Newman et al., 2010), C_{UMass} (Mimno et al., 2011), and C_{NPMI} (Bouma, 2009b).

A common explanation of the used variables in all the following coherence equations is as follows:

- i and j are integer indices.
- The notation $\sum_{i<j} log$ means that the sum is taken over all pairs of words such that the index i is less than the index j. This ensures that each pair of words is considered only once. For instance, if there are three words in a topic, the pairs would be (*word1*, *word2*), (*word1*, *word3*), and (*word2*, *word3*).
- w_i refers to the word in the topic corresponding to the index i.
- w_j refers to the word in the topic corresponding to the index j.

The details of each coherence equation are as follows:

a) *C_V Coherence Score*. C_V measures evaluate texts that machine learning models generate. For topic modeling, the C_V measure deals with the indirect coherence between words in each individual topic. This measure combines Normalized Pointwise Mutual Information (NPMI) and cosine similarity. The indirect similarities mean that even though some terms rarely occur together, they belong to the same topic. The general formula of topic coherence is defined as:

$$coherence = \sum_{(i<j)} score(w_i, w_j)$$

where both w_i,w_j are the top terms of each topic.

b) *C_{UMass} Coherence Score*. C_{UMass} measure is based merely on the co-occurrence statistics from the specific dataset of documents rather than the external reference corpus (Mimno et al., 2011). The C_{UMass} metric is intrinsic because it ranks each word in the list with its predecessors and successors (Fu et al., 2021). This metric uses the pairwise score function, and it is calculated by:

$$C_{UMass} = \frac{2}{N \cdot (N-1)} \sum_{i=2}^{N} \sum_{j=1}^{i-1} log \frac{p(w_i, w_j) + \epsilon}{p(w_j)}$$

c) *C_{UCI} Coherence Score*. C_{UCI} measure is based on Pointwise Mutual Information (PMI). This measure makes the most coherence with human judgments since it does not rely on the given corpus. UCI coherence score is calculated by:

$$C_{UCI} = \frac{2}{N \cdot (N-1)} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} PMI(w_i, w_j)$$

where $PMI(w_i, w_j)$ is defined as:

$$PMI(w_i, w_j) = log \frac{p(w_i, w_j) + \epsilon}{p(w_i) \cdot p(w_j)}$$

d) *C_{NPMI} Coherence Score*. C_{NPMI} is considered an extension of the C_{UCI} because it uses the normalized PMI instead of the regular PMI (Aletras & Stevenson, 2013; Lau et al., 2014). This measure produces the most considerable correlation to human topic coherence evaluation (Bouma, 2009b). The authors (Lau et al., 2014) reported the superior performance of this metric, and it is calculated by:

$$C_{NPMI} = \frac{1}{\binom{N}{2}} \sum_{j=2}^{N} \sum_{i=1}^{j-1} \frac{log \frac{p(w_i, w_j) + \epsilon}{p(w_i)p(w_j)}}{-log\, p(w_i, w_j) + \epsilon}$$

3.5. Topic Modeling Libraries and Toolkits

a) *Stanford Topic Modeling Toolbox (TMT) (Stanford University, 2009)*. This toolkit helps to perform analysis on the textual dataset. It has the ability to import and modify text from Excel to train topic models (LDA, LLDA), select parameters, and generate rich outputs.

b) *Mallet (McCallum, 2002)*. This package is used for unsupervised topic modeling and document clustering.

c) *Gensim (Rehurek, 2011)*. This toolkit allows the estimation of the LDA model from a training corpus. It helps to develop LDA-extension topic models, and it also performs topic distribution inference on new documents. It was originally developed on Cython.

d) *pyLDAvis (Sievert & Shirley, 2015)*. This package is mainly used for the visualisation of the generated topic. It gives a great understanding by interpreting the topic of a topic model.

e) *Tomotopy (Choi, 2019)*. A very high-performance library compared to others. It was originally built based on C++ and had a Python extension, including several pre-developed topic models.

f) *MUSE (Conneau & Lample, 2021)*. It is a Python library that helps to apply word embeddings. It provides rich and high-quality dictionaries, and it has the option to train the model based on CPU and GPU.

g) *BERTopic (Grootendorst, 2022)*. The Python toolkit for topic modeling uses BERT embeddings and class-based TF-IDF to construct dense clusters. It supports both supervised and unsupervised topic modeling.

h) *Natural Language Toolkit (NLTK) (Bird et al., 2009)*. It is a very common library for NLP processes and tasks in general. It helps to preprocess textual data for topic modeling.

i) *Tmtoolkit (WZB, 2019)*. It is an easy-to-use Python library that supports various languages to apply topic models in social sciences and journalism.

3.6. Benchmark Models

Benchmarking is a critical process in scientific research that allows for comparing different models or methods under the same conditions. It provides a standardized performance measure, enabling researchers to understand the strengths and weaknesses of various approaches. This is particularly important to investigate their performance on web content data and evaluate the proposed topic model in the following chapters.

The selection of the benchmark topic models in this study follows the following criteria:

a) *Aim of the study*. The first step is to choose topic models that would facilitate achieving the aim of this study, which is to classify web content data. Therefore, this study considers the benchmark models under the web content analysis category (refer to subsection 3.2.1).
b) *Model popularity*. The study focuses on selecting the most common and well-known models in the field to ensure a robust community for support, well-maintained software implementations, and a rich literature base for result interpretation.
c) *Model complexity*. Selecting models with a different range of complexity levels to allow for a more comprehensive evaluation of performance across different scenarios. Simpler models may perform well on less complex tasks or smaller datasets due to their efficiency and fewer assumptions, while more complex models may be necessary to capture intricate patterns in larger or more complex datasets. Therefore, including models of varying complexity in a benchmark ensures a more robust and generalizable understanding of model performance.

d) *Model novelty*. Choosing topic models with strong novelty allows improved performance and diversifies the range of evaluated methods. Various novelties, such as topic correlations, hierarchical structure, and supervision, also comprehensively overview the results.

Given these criteria and based on the state-of-the-art overview of the previous sections, Table 3.1 summarises the characteristics and limitations of this study's chosen benchmark topic models.

Table 3.1: Characteristics and limitations of the benchmark topic models

Model	Characteristics	Limitations
LDA (Blei et al., 2003)	Requires manual removal of stop-words.	Inability to model relations among topics
	Previous studies have found that the representation of the relationships among topics is out of LDA's scope.	The number of topics (K) must be known. Failure in the face of a large number of vocabularies
HLDA (Griffiths et al., 2004)	The system discovers topics within a corpus hierarchically, placing abstract terms at the base of the hierarchy and locating detailed and specific terms near the leaves of the hierarchy.	Ignoring lexical co-occurrence and showing poor consideration for word dependencies, the system's performance slows down with increased hierarchy levels, leading to long execution times.
DMR (Mimno & McCallum, 2008)	Uses Gibbs sampling and provides inferences about hidden variables.	The tendency to underestimate abundant features and overestimate marginal features results in a more complicated sampling distribution (low efficiency)

CTM (D. Blei & J. Lafferty, 2006)	Uses a normal logistic distribution to create relations among topics. Allows the occurrences of words in other topics and topic graphs.	Requires lots of calculation. Results in lots of general words inside the topics.
PTM (Zuo et al., 2016)	Analyses topics without using auxiliary contextual information assumes each short text relates to only a single pseudo document and avoids overfitting when the training corpus is relatively scarce.	The system is unable to directly apply to raw input data, requiring heuristic methods to enrich the data, and it generates high-frequency but topic-irrelevant words while also struggling to deal with extremely sparse and noisy data.
sLDA (Mcauliffe & Blei, 2008)	Assigns a label on each training document (in distinction from the LDA model) and offers improved predictions over regressions on words alone. Applicable, besides text, on social networks image classification.	Marking documents with a response variable is required but unable to use for multi-class classification problems, and its application to large datasets is labour-intensive and expensive due to the labelling process.

3.7. Summary

Topic modeling has become an essential technique and has been utilized in various applications, including web content mining tasks. This chapter categorizes the statistical distribution models used in state-of-the-art topic models and taxonomizes them based on their web application utilisation. The critical review of these taxonomies reveals a significant gap. While previous studies have extensively explored and evaluated various topic modeling techniques, their application and evaluation in the context of web content classification and filtering have been lacking. This study addresses this gap by applying these techniques to the problem of web content classification and filtering and evaluating their performance in this context. The evaluation of benchmark topic models, as Section

5.3 will demonstrate, emphasizes the need for a more coherent topic model specifically designed for web content data.

In summary, this chapter established the following facts and findings from the state-of-the-art topic modeling field:

a) Probabilistic models, including distribution models, are highly effective and versatile tools due to their accurate probability estimation of unseen data, capturing a wide range of data patterns and incorporating prior knowledge.
b) Among the categories of topic models, web content analysis provides a robust and nuanced approach to classifying web content data.
c) The main challenge of the state-of-the-art topic models utilized for web content analysis is neglecting the structure of the textual contents within the HTML tags of web pages.
d) Among topic models that analyze web content, this chapter benchmarks the CTM, DMR, LDA, HLDA, PTM, and sLDA due to their solid baselines, wide usage, and relativeness.
e) Python is the primary programming language that facilitates designing, developing, and visualizing topic modeling with the help of Gensim, NLTK, and pyLDAvis.

This chapter defined the benchmark topic models and highlighted some important issues with topic modeling in analyzing web content. Addressing the issues and inheriting the advantages of these models constitute a roadmap to building a useful and coherent topic model for web content data. Thus, developing an efficient web classification framework using the coherent topic model, as the subsequent chapter will discuss in detail.

CHAPTER 4: THE CYBER PARENTAL CONTROL FRAMEWORK USING WEB CONTENT TOPIC MODELING

This chapter presents comprehensive details of the main contribution of the study. It first addresses the proposed HTML Topic Model (HTM) for learning interpretable topics in web content data. The HTM topic model is based on the LDA and aims to enhance the performance of topic modeling for web content-based data. The chapter illustrates the formulation and notation, the generative process, and the mathematical representation of the HTM topic model. The chapter then addresses the proposed web classification framework for cyber parental control. The framework employs a multistep approach that uses the URL blacklist and whitelist approaches as the first and second filter layers and the HTM in the final classification layer, as Figure 4.1 demonstrates. The final layer includes several modules to scrape the webpage and preprocess its content data to be applied to the HTM model, as the following subsections illustrate. The framework classifies and filters objectionable web pages based on their content data by using the framework layers and modules, achieving the main aim of this study. Finally, to evaluate and validate the framework and its layers, the chapter provides a detailed description of the collected datasets and the steps of creating the ground truth dataset.

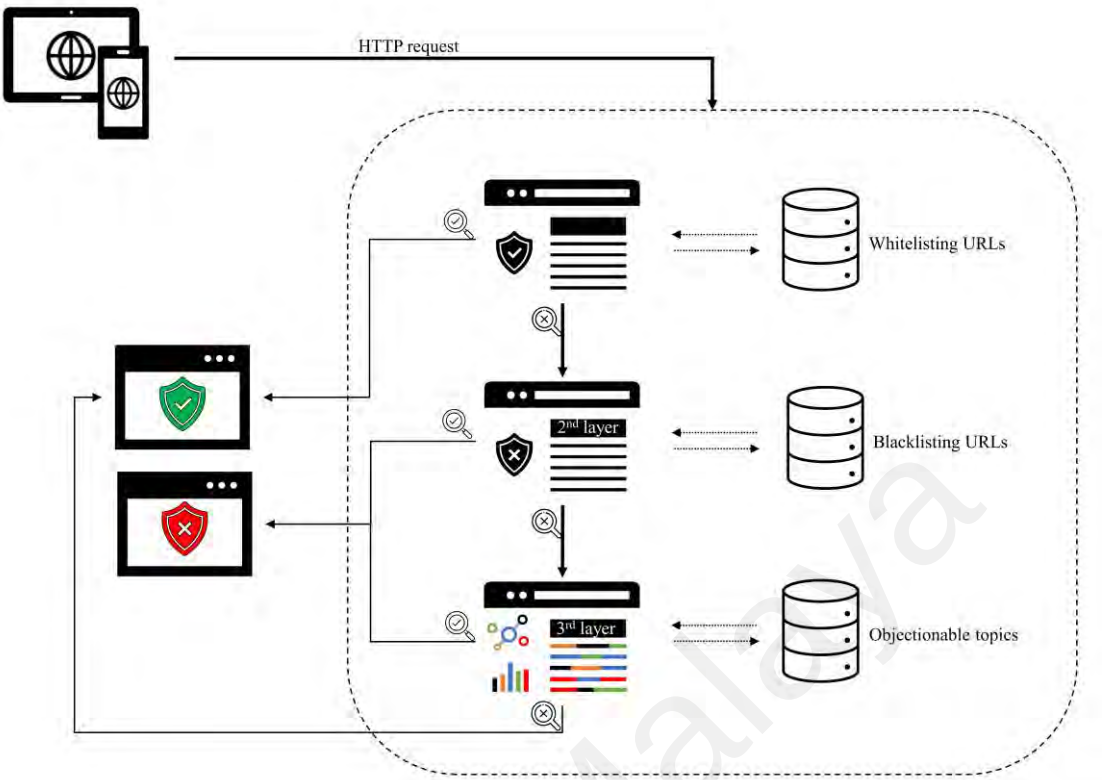

Figure 4.1: Conceptual architecture of the cyber parental control framework using web content topic modeling

4.1. HTML Topic Model

A webpage represents web contents in a hypertext document provided by a website, usually comprising several web pages. HTML tags represent web pages, constituting their hyperlinked structures and textual contents. These tags, such as <*title*>, <*metadata*>, <*a*>, <*b*>, <*ul*>, <*li*>, <*hr*>, and <*img*> normally contain very short textual contents. Unlike conventional text documents, combining these textual contents from these tags results in a sparse and incoherent document. The sparseness and incoherence create challenges and cause the traditional text mining and topic model methods to be ineffective for web content mining (Figueiredo et al., 2013).

The HTM topic model considers the structure of the textual contents within the HTML tags to extract topics from a webpage. HTML tags are usually used to add textual content to the webpage. The HTML tag element consists of a start tag, end tag, attribute name, attribute value, and textual content, as Figure 4.2 illustrates. The HTM topic model

considers all HTML tags that contain visible textual content. In general, the HTM model extracts only the visible textual content of each HTML tag element of the webpage and uses it as a document within a webpage and each webpage in a website as a document. The extracted topics of these tags' textual content can describe the webpage efficiently and, therefore, create practical web topic modeling. Besides that, HTML attributes that provide additional information with visible textual content are also considered. These attributes are alt, title, label, value, placeholder, and data-*attributes. The following subsection describes the generative process of the HTM model.

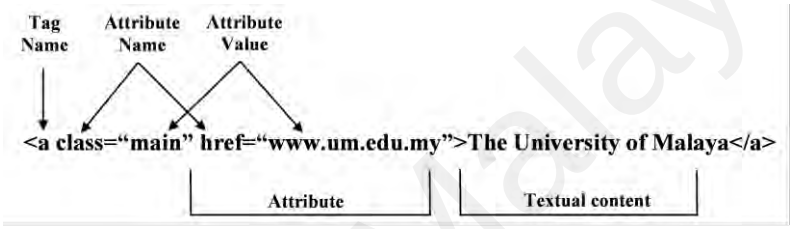

Figure 4.2: HTML tag elements

4.1.1. Problem Formulation and Notation

This section describes the problem formallyaddressed and illustrates the used notations in the following subsections, and establishes the mathematical or symbolic language to be used throughout the proposal. It also sets the foundation for understanding the model's structure and functionality, thereby facilitating comprehension and further development. The definition of the problem is as follows:

Consider a collection of web pages defined as follows:

$$Dataset\ (D) = \{WP_0, WP_1, \dots, WP_{p-1}\} \quad \textbf{(1)}$$

where WP_i is the i-th webpage of a dataset collection D, and p is the number of web pages in the collection. Each of these web pages is composed of HTML tags as follows:

$$Webpage\ (WP) = \{TG_0, TG_1, \dots, TG_{t-1}\} \quad \textbf{(2)}$$

where TG_i is the i-th HTML tag of a webpage WP and t is the number of HTML tags in the webpage.

An HTML tag topic in a given webpage is the distribution of all words relating to this webpage and can be represented as,

$$\theta_{tg} = \left\{\theta_{tg_i}\right\}_{i \in 1_{tg}} \sim Dir(\cdot \,|\alpha_{wp}) \quad \textbf{(3)}$$

Taking an education news webpage as an example, which includes many HTML tags, each may include different sub-topics. However, some tags in the webpage may include some other recent news and the side of the webpage for users to read. This news can be related to education as well as other topics such as political news, health news, and many others. In this case, taking the webpage as one piece could give low topic coherence and, therefore, generate low topic quality. The webpage topic modeling problem aims to find topics occurring on a webpage and ensures that the generated topics are semantically coherent.

Before introducing the generative process and the mathematical explanation of the model, Table 4.1 tabulates the used notations.

Table 4.1: Description of the used symbols in the HTM topic model

Symbol	Description
α	Per-document topic distributions
β	Per-topic word distribution
θ_1	Topic distribution for TG
θ_2	Topic distribution for WP
φ	Word distribution for T
Z	Topics of the n-th word in TG
W	Specific word
V	Set of words in the vocabulary
WP	Webpage
TG	HTML Tag

N	Number of words in a given document

4.1.2. The Generative Process

Like the LDA topic model, the HTM model is based on a generative statistical model, and it uses latent factors to capture the semantic similarities of words and documents. The generative process of the HTM model is as follows. Firstly, there is a need to specify the optimal number of topics represented by (*K)*. Then, randomly choose a distribution over topics (a multinomial of length K). A specific webpage (WP) is modeled as a sequence of words $WP = (WP_1, \dots, WP_\iota)$ of length $\iota \sim Poisson(\xi)$, where ξ is pre-specified. For this webpage WP, a K-dimensional probability vector θ with non-negative coordinates summed to one is used to model the topic mixture. Three probability distributions are assumed to be multinomial distributions: $p(z|wp)$, $p(z|tg)$, and $p(w|z)$. Therefore, the topic distributions in all web pages share the common Dirichlet prior α, and the word distributions of topics share the common Dirichlet prior β. Given α and β as the parameters for webpage wp, parameter θ_{wp} of a multinomial distribution over K topics is generated from Dirichlet distribution $Dir(\theta_{wp}|\alpha)$. Similarly, parameter θ_{tg} of a multinomial distribution over K topics is generated from Dirichlet distribution $Dir(\theta_{tg}|\alpha)$. For topic t, the parameter φ_t of a multinomial distribution over V words is derived from Dirichlet distribution $Dir(\varphi_t|\beta)$. As a conjugate prior for the multinomial, the Dirichlet distribution is a convenient choice as a prior and can simplify the statistical inference in the HTM model. The likelihood is multiplied through all the web pages and maximized with the technique of variational inference for the estimation of α and β.

A summary of the generative process for a set of web pages is as follows:

For each topic $t \in \{1, \dots, T\}$

Generate $\varphi_t = \{\varphi_{tw}\}_{w=1}^{V} \sim Dir(\cdot|\beta))$

For each webpage $WP \in \{1, \dots, N\}$

Generate $\theta_{wp} = \left\{\theta_{wp_i}\right\}_{i \in 1_{wp}} \sim Dir(\cdot \mid \alpha_{wp})$

For each HTML tag tg in the webpage WP

Generate $\theta_{tg} = \left\{\theta_{tg_i}\right\}_{i \in 1_{tg}} \sim Dir(\cdot \mid \alpha_{wp})$

For each word w in the HTML tag tg

Generate $z_{tgn} \in \left\{\theta_{tg_i}\right\}_{i=1} \sim Multinominal(\cdot \mid \theta_{tg})$

Generate $w_{tgn} \in \{1, \ldots, V\} \sim Multinominal\left(\cdot \mid \varphi_{z_{tgn}}\right)$

The following snapshots elaborate more on how a webpage will transform from metadata to preprocessed data ready to be inserted into the HTM topic model. Figure 4.3 shows a snapshot of the webpage metadata. Notice that the metadata contains various HTML tags, such as , , <svg>, <p>, and <h4>. The HTM assumes that the webpage is a distribution of tags; therefore, each tag will be preprocessed separately. Notice that only visible text tags will be used, and their textual content will be extracted, as explained in Figure 4.3 (snapshot of an unobjectionable webpage's HTML content).

Figure 4.3: Raw HTML content of a webpage

Figure 4.4 shows a snapshot of the preprocessed textual data, where each tag is represented separately as a list. The following section represents these steps mathematically and elaborates on the role of the tags of a webpage using a plate notation of the HTM topic model.

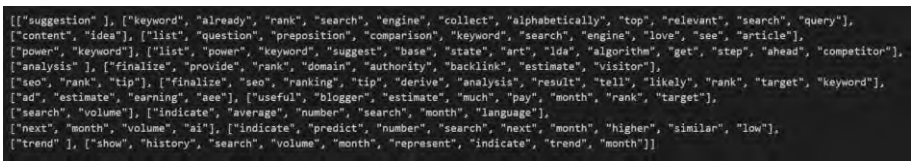

Figure 4.4: Pre-processed HTML content of a webpage

4.1.3. Mathematical Model

The HTM topic model, like the LDA model, is based on the probability distribution model. Figure 4.5 shows the graphical model of the HTM topic model, referred to as the plate notation graph.

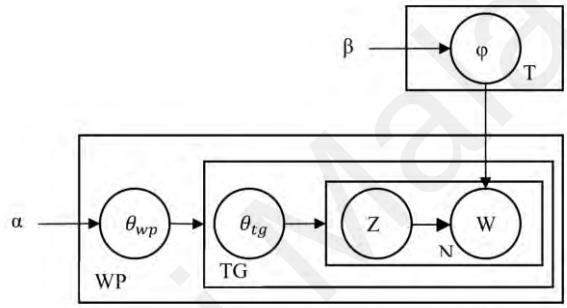

Figure 4.5: Plate notation of the HTM topic model

The model infers the distribution of the hidden variables by using the joint probability distribution as follows:

$$\left(\underset{topics}{\beta}\ ,\ \underset{proportions}{\theta}\ ,\ \underset{assignments}{\mathrm{Z}}\ \middle|\ \underset{Webpages}{WP}\right) \qquad (4)$$

hidden variables — evidence

This inference aims to approximate the posterior $\rho(\beta, \theta, \mathrm{Z}|W)$ with the distribution $q(\beta, \theta, \mathrm{Z})$ using the variance inference, simplifying the model analysis. Figure 4.6 illustrates the inner plate representing the probability distribution of words per topic to simplify the model.

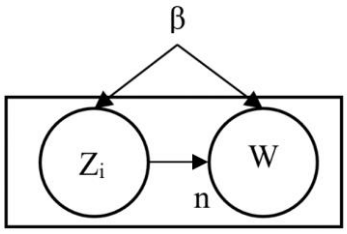

Figure 4.6: Inner plate notation of the HTM topic model

In this sub-graph, β acts as a global variable, while $Z|W$ acts as a local variable for each word in the corpus. This part is inherited as it is from the LDA model. The mathematical definition of this plate is as follows:

$$p(\beta, Z_{1:n}, W_{1:n}) = p(\beta) \prod_{i=1}^{n} p(Z_i|\,\beta)\, p(W_i|Z_i, \beta) \qquad (5)$$

This sub-graph is associated with the per-HTML tag topic proportion variable and the word distribution for each topic. Figure 4.7 illustrates this association of the model.

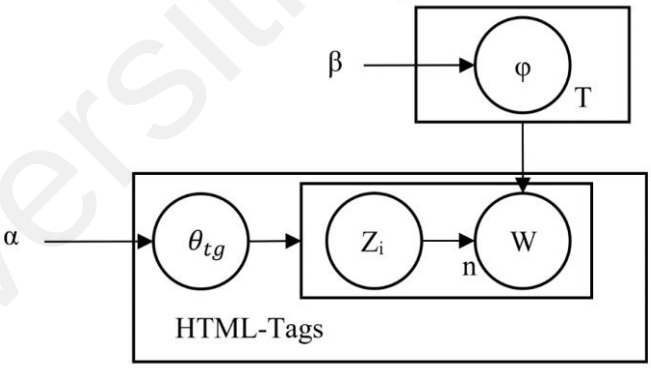

Figure 4.7: Outer plate notation of the HTM topic model

The parameter α of the Dirichlet distribution models the topic distribution variable θ_{tg} per HTML tag, while the parameter φ of the multinomial distribution models each specific associated topic Z_i. This association is defined as:

$$q(\beta, \theta_{tg}, Z) = \prod_{k=1}^{K} q(\beta_k|\varphi_k) \prod_{tg=1}^{TG} q(\theta_{tg}|\alpha_{tg}) \prod_{n=1}^{N} q(Z_{tg,n}|\varphi_{tg,n}) \qquad (6)$$

Once the HTM model processes HTML tags, the model then applies a similar step on all the given web pages. The following equation describes the process as follows:

$$q(\beta, \theta_{wp}, \theta_{tg}, Z)$$

$$= \prod_{t=1}^{T} q(\beta_t|\varphi_t) \prod_{wp=1}^{WP} q(\theta_{wp}|\, \alpha_{wp}) \prod_{tg=1}^{TG} q(\theta_{wp,tg}|\alpha_{wp,tg}) \prod_{n=1}^{N} q(Z_{tg,n}|\varphi_{tg,n}) \quad (7)$$

Once the HTM model processes all the web pages, the model then updates the parameters of the topics ($\varphi \; and \; \alpha$). The model updates these parameters after each iteration. In each iteration, as the α_{tw} value increases, the chance of selecting the word W from the HTML tag TG in topic T also increases.

4.2. Cyber Parental Control Framework

The proposed cyber parental control framework aims to filter objectionable web content and provide online safety for children. This section first illustrates the proposed cyber parental control framework, combining URL whitelisting, URL blacklisting, and content-based filtering approaches. The framework consists of the following three layers:

a) *Whitelisting Layer*. The initial layer of this framework employs a whitelist technique, whereby the target URL is juxtaposed with a predetermined list of unobjectionable URLs within the framework's database. If the webpage's URL corresponds with any on the whitelist, access is subsequently granted. If no correlation is discerned, the webpage is relegated to the second layer for additional analysis. A prominent advantage of this layer resides in its expeditious classification process, which starkly contrasts with the time-intensive nature of content-based classification methods. Nevertheless, this technique is not devoid of challenges. Maintaining an accurate, updated whitelist poses a considerable hurdle. The framework incorporates a unique strategy to tackle this issue: commencing with a vacant whitelist and appending URLs

only after they have been substantiated as unobjectionable via the framework before, thus fostering a dynamic, evolving whitelist.

b) *Blacklisting Layer*. The second layer operates analogously to the first, yet its focus is pinpointed on detecting objectionable web pages. This layer applies a blacklist technique, comparing the target URL with a pre-established list of objectionable URLs. Should the webpage's URL align with any entry in the blacklist, access is consequently denied. If no alignment is established, the webpage is dispatched to the third layer for further analysis. Similar to the first layer, the blacklisting layer's strength lies in its rapid classification capability. However, this layer, too, contends with the issue of maintaining an accurate, updated blacklist. The framework adopts a similar strategy to the first layer to counteract this issue: commencing with an empty blacklist and appending URLs only after they have been confirmed as objectionable via a series of filters, thereby creating a dynamic, evolving blacklist.

c) *Topic Modeling Layer*. When the webpage is not classified in the first two layers, the last layer performs a content-based classification based on the HTM topic model. This layer extracts the topic of the webpage by applying topic modeling methods on textual content and compares the webpage’s topic with the trained classifier model.

The topic modeling layer contains web scraping, topic modeling, and classification modules, as Figure 4.8 illustrates. The sections explain these modules' aims, steps, and utilized tools.

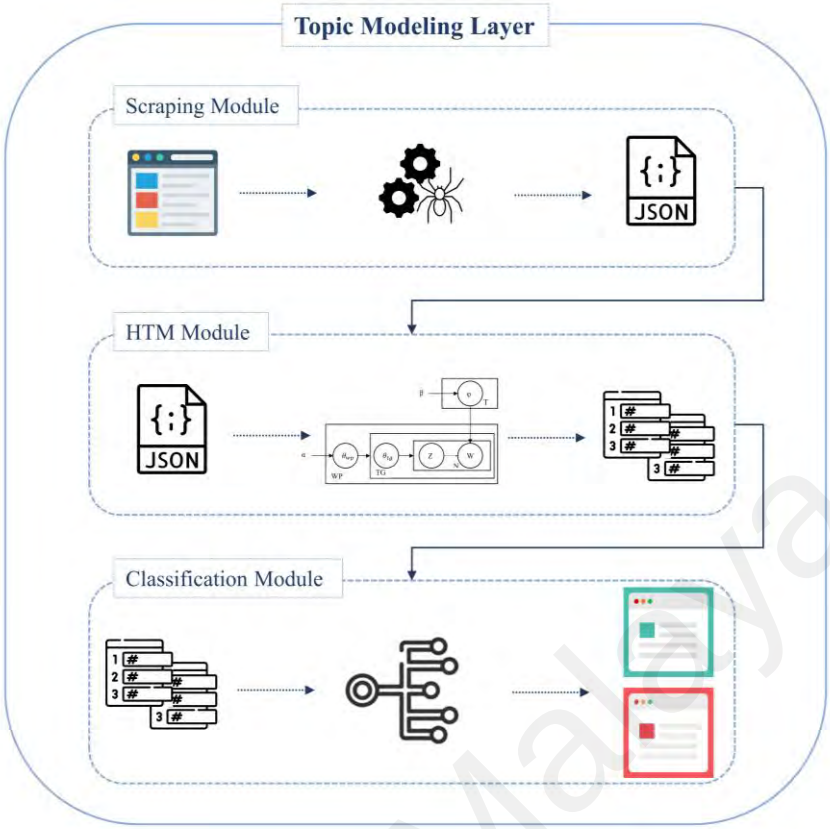

Figure 4.8: Modules of the topic modeling layer

4.2.1. Scraping Module

The scraping module, also known as web data extraction or web harvesting, is a technology that simulates human web browsing to collect large amounts of data from various websites. This data is typically stored in external storage mediums such as cloud services, databases, or JSON files. The advantage of a scraper lies in its speed and automation capabilities, which significantly reduce the time and effort required for manual data collection. While web scraping usually involves two components - web crawling and data extraction - in this framework, the scraping module receives a specific webpage URL for classification, eliminating the need for the crawling component. Therefore, the module's focus is on extracting the textual content data from the requested webpage, a process that involves six steps as follows:

a) Validate the requested URL.

b) Inspect the webpage and extract HTML.

c) Parse the HTML content.

d) Detect visible textual data by their specific TAGs.

e) Store the extracted visible textual content into a list of TAGs.

f) Export the extracted list using JavaScript Object Notation (JSON) format.

Various programming languages, such as Java, JavaScript, PHP, and Python, are utilised to develop and implement web scraper tools. The advantages of utilizing Python in developing this module are as follows:

a) *Libraries*. Has the greatest number of libraries and modules for scraping web pages among these languages (Mahto & Singh, 2016; Thivaharan et al., 2020).

b) *Optimization*. Decrease the size of the code, resulting in faster outputs and more efficient results, and make the code much simpler and more user-friendly (Pratiba et al., 2018).

c) *Community*. Including built-in capabilities for acquiring data directly from a website and having a large community base. This study uses Python to construct this module because of these advantages.

This module mostly utilises BeautifulSoup (BeautifulSoup Library, 2015) for webpage scraping and source code analysis. BeautifulSoup is an open-source library that includes the capacity of architecture binding with the Document Object Model by default (DOM) (Mahto & Singh, 2016; Soup, 2020). It generates a tree-like structure of a web page's content for navigation and content extraction. This module utilised several helpful libraries, like *URLLib*, *Top-Level Domain (TLD)*, and *TLDExtract*, to obtain the HTML of a webpage.

The module's final result is a JSON file format containing the list of all visible textual tags of the target webpage. This file will be used as input to the topic modeling module.

4.2.2. Topic Modeling Module

This module discovers the topics of the target webpage by utilizing the HTM topic model. Implementing the HTM topic model is in Python with the help of the *Gensim* library. This module includes several tasks to perfume its results.

a) *Import data*. The first task is to import the data from the scraping module. The data is represented in JSON file format and contains visible textual data of the webpage.

b) *Preprocess data*. The second task is to clean and preprocess the data. Data preprocessing is an essential step in machine learning and data mining in general. This task assures the quality and clarity of the resulting topics for topic modeling. In order to transfer the selected documents into meaningful and formatted data, this task consists of four steps which are explained as follows:

- *Text tokenization*. Tokenization is the action of splitting the text into sentences and the sentences into words. Words are then lowercase, and punctuation marks are removed.
- *Stopwords removal*. Stopwords are English words that do not add much meaning to a sentence. They can safely be ignored without sacrificing the meaning of the sentence. This step also includes removing all special characters and words, such as email signs, newlines, and quotes.
- *Bigram constructing*. A Bigram, a particular form of n-gram with two adjacent elements, is a probabilistic model that aims to predict the different meanings of words when they are in a sentence. This step is essential because sometimes word groups are more beneficial in explaining the meaning than single words.
- *Word lemmatization*. Lemmatization, a special case of normalization, aims to reduce the inflectional forms and sometimes derivationally the related forms of a word to a common base form. This step maintains the part-of-speech tagging.

c) *Generating topics*. After data preparation for the HTM topic model is done, the main task is to apply the data to the developed HTM topic model. The main parameters used to build the HTM topic model are the corpus in a Bag-of-Words (BoW) format, Word Identification (id2word) mapping to the dictionary, and the number of topics. Other parameters such as alpha, random state, chunk size, and passes are set to default values.

The trained HTM topic model will then generate a vector of topics for the webpage as the output of this module. These vectors are exported as a list for the classification module.

4.2.3. Classification Module

Once the topics have been extracted from the webpage, one of the distinguishing features of the proposed framework for cyber parental control is an automatic classification of the webpage into the category of unobjectionable or objectionable content.

This module primarily uses supervised machine learning algorithms to classify web pages and decide whether they include objectionable topics. This module provides the classification of the target webpage and is based on the topic vector features retrieved by the module that came before it. In the classification module, this study conducted comparative experiments using the SVM, NB, RF, LR, and KNN classifiers to validate the effectiveness of the cyber parental control framework. This module begins by training these classifiers on our publicly available ground-truth dataset and then constructs a trained model that is capable of reliably classifying web

pages. The output of this module is either a 0 or a 1 for the classification (0 for the unobjectionable webpage, 1 for the objectionable webpage).

This module is implemented using Python programming language. The implementation utilized the scikit-learn library for building the classifiers. Scikit-learn is an open-source machine learning software based on Python (Pedregosa et al., 2011). It includes most machine learning algorithms. This module also utilised several helpful libraries, like *Pandas* and *Numpy*, to classify the webpage.

4.3. Operational Characteristics

The Operational characteristics of the proposed cyber parental control framework include the following:

a) *Merged learning*. The proposed framework merges unsupervised and supervised machine learning models, which brings many advantages. Among the advantages are improving performance, enhancing interpretability, and pre-training.

b) *Scalability*. Using the proposed HTM topic model, the framework is able to handle large numbers of web pages and deal with the computational complexity of topic modeling. However, the scalability of the framework also relies on computing resources.

c) *Robust and practical*. The framework uses the proposed topic model to perform well on unseen and new web pages, even if they include different topics. This characteristic can be seen in the framework's coherence and accuracy scores in Section 5.4 and Section 5.5. A few factors helped the framework's robustness, including data variability, pre-processing, and the ensemble of models, as Chapter 5 illustrates.

d) *User-friendly and flexible*. The input and output are simple and straightforward, hiding the complexity of the framework and its layers. The input is the URL of the target webpage, and the output is the label of the webpage (objectionable or unobjectionable). Besides that, the module-based design ensures easy maintenance as well as provides flexibility for future extensions.

4.4. Dataset

The current studies of filtering objectionable web content evaluated their models and frameworks based on inconsistent datasets. There is a lack of a standard dataset in the current web content filtering studies, as shown in the literature review of this study (Table 2.6 in Section 2.4). Most studies design and build their dataset to suit their model or framework. Moreover, few studies built interesting datasets (Altay et al., 2019; Rao et al., 2020; Sahingoz et al., 2019). However, these datasets focus only on a partial topic of objectionable topics such as phishing, malicious, spam, hate, violence, and pornography.

For this reason, these datasets are unsuitable for the proposed cyber parental control framework. Table 2.6 also shows that only (Hussain et al., 2018; Patel et al., 2015; Rajalakshmi et al., 2020; Zhao et al., 2018) are suitable datasets; however, none is publicly available. Given these factors, there is a need to create a dataset that contains objectionable and unobjectionable websites.

Each experiment in this study uses a different dataset for evaluation. This study uses three different datasets explained as follows:

a) *Dataset I – Conventional document data*. This dataset contains article-based documents from Wikipedia. The second experiment uses this dataset to compare the benchmark topic models' performance.
b) *Dataset II – Web-based content data*. This dataset contains extracted web content data from more than 1,000 websites. The second experiment uses this dataset to compare the benchmark topic models' performance. The third experiment also uses this dataset to evaluate the performance of the proposed HTM topic model against the benchmark topic models.
c) *Dataset III – Objectionable web content ground truth*. This ground truth dataset contains 7,000 labelled websites, with 3,500 objectionable websites and 3,500

unobjectionable websites. The fourth experiment relies on this dataset to train the classifier models and evaluate the accuracy of the proposed cyber parental control framework.

Figure 4.9 illustrates the methodology used for each experiment's different datasets and its preprocess. The common process of all datasets starts with data collection, extraction, and preprocessing. For the second experiment (section 5.3), this study uses two sources of data to create two datasets; conventional document-based (*Dataset I*) and web content-based (*Dataset II*) datasets. The third experiment (5.4) uses only the web content-based dataset (*Dataset II*). The fourth experiment contains a few more processes to create the ground-truth dataset (*Dataset III*) for evaluating the cyber parental control framework. The following subsections address these preprocesses and each dataset in detail.

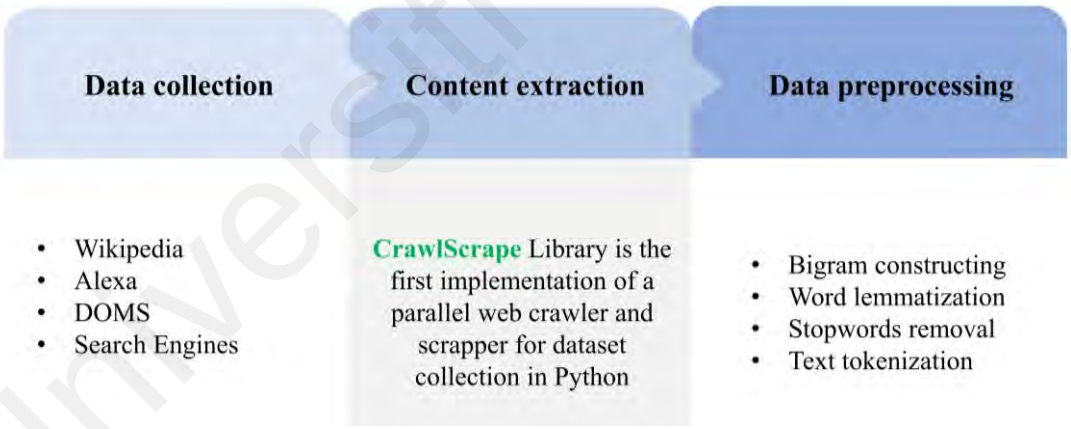

Figure 4.9: The methodology of the used datasets in this study

4.4.1. Data and Methods

This section describes the common processes of collecting web pages and extracting and pre-processing data. Both *dataset II* and *dataset III* (as shown in Section 4.4.3) rely on these processes, which are detailed as follows:

a) *Webpage collection*. This study focused on data collection methods for web content, opting for a hybrid approach that blends manual and automated means. Data were

gathered from various online sources, including the Alexa dataset, search engines such as Yandex and Google, and links from external web pages. Each source provided a classification of the websites into different categories. The study classified these websites as either objectionable or unobjectionable based on this categorization. For search engines, the classification was based on the keywords used in the search query. Websites associated with keywords like "porn", "erotic", "gambling", and so on were deemed objectionable. The sources of these websites are outlined in Table 4.2.

Table 4.2: Website collection sources and number of websites

Source	Objectionable sites	Unobjectionable sites	Total
Alexa	0	1,500	1,500
DMOZ	1,500	1,000	2,500
Google	500	500	1,000
Yandex	500	500	1,000
Yahoo	500	500	1,000
Internal links	1,000	0	1,000
Total	**4,000**	**4,000**	**8,000**

b) *Web content extraction*. This study utilized a process of web crawling, scraping, and parsing to extract the content of various websites. Web crawling indexed all the web pages within a particular website by methodically browsing the World Wide Web. The HTML scraping extracted vital elements like paragraphs, images, bold text, titles, and metadata. This study designed and developed a specific Python library to facilitate this process that integrates *BeautifulSoup*, *LXML*, *MechanicalSoup*, *Requests*, *Scrapy*, and *Urllib* libraries. This library, named *"CrawlScrape"*, with its source code, is publicly available on GitHub (Altarturi, 2022), and its development details are addressed in Section 4.4.2 of the study. In order to guarantee the ethical compilation of this step, the procedure of scraping and extracting content meticulously followed Gab's Robots Exclusion Protocol (specifically, the robots.txt file), its privacy policy,

and its Terms of Service. Notably, none of these documents impose limitations on web-scraping activities.

c) *Data pre-processing*. Data preprocessing is an essential step in machine learning and data mining in general, and this task assures the quality and clarity of the resulting topics. In order to transfer the selected documents into meaningful and formatted data, this task consists of four steps: text tokenization, stop-word removal, bigram constructing, and word lemmatization. These steps are the same as the preprocessing steps of the proposed HTM topic model (as shown in Section 4.2.2). Algorithm I illustrates the task of data preprocessing for topic modeling.

Algorithm I: Data preprocessing task for web topic modeling

```
Import ntlk, ntlk.corpus.stopwords, gensim.models.Phrases, spacy
For each webpage_extracted_content
    Remove special characters (email signs, newlines, quotes)
    For each sentence
        Apply tokenization
    Load English stop_words
    For each sentence
        Remove stop_words
    Create bigram_model
  For each word
      Apply bigram_model
  Apply lemmatization
  Store and Return lemmatized_doc
```

4.4.2. CrawlScrape Library

This study designed and developed the CrawlScrape Python library, the first implementation of a parallel web crawler and scrapper in Python. CrawlScrape is an open-source Python library for the solution of efficient and easy web crawling and data scraping for dataset collection. Developers and researchers may use this library for data collection and indexing. This library provides an efficient, simple, extensible, and parallel implementation of crawling and scraping a bulk of websites. It combines web crawling

and scraping to extract the target websites' features easily and efficiently. The library also uses a multithreading approach to improve throughput and minimize system resource usage.

The library allows users to specify a list of target websites to be crawled and scraped, along with a few other parameters, as Table 4.3 enumerates.

Table 4.3: The parameters details of the CrawlScrape library

Parameter	Details
dataset	A string list of target websites
saving_directory	A string of the directory where extracted data will be saved By default, the saving directory will be at save the code root directory (root directory/Crawled Dataset/)
max_crawling_number	Maximum number of crawling internal URLs of each website By default, 250 web pages (internal URLs)
collection_source	The source of collecting the websites, if applicable By default, Null
label	The label/category of the websites, if applicable By default, Null
sub_label	The second level label/category of the websites, if applicable By default, Null
crawl_time_out	The timeout of crawling and scrapping each website in seconds By default, 7200 (2 hours)

The crawler begins from each website URL and progressively extracts the internal URLs (web pages) that belong to the website. These internal URLs are added to the frontier list to be processed. First, the crawler initiates the multithreading working environment, comprising the index structure, the repository containing the collection of web documents, and the cluster nodes (workers) for parallel computing using multithreading. Multiple worker threads perform crawling, and the work-pool-handler

component prepares a pool of URLs to be processed in parallel. Then, each worker executes the following functions for the given URL:

a) Retrieve the HTTP source code of the webpage using a GET request.

b) Parse and extract all links contained in the HTML tags.

c) Detect and store internal URLs that belong to the websites.

d) Parse and extract all visible textual content in the HTML code.

e) Parse and extract all visual source (URL) content in the HTML code.

f) Extract other features of the webpage.

As a result, for each URL, each worker stores a JSON formatted file containing all extracted HTML document information and HTTP header details. Figure 4.10 illustrates the architecture and main components of the CrawlScrape library.

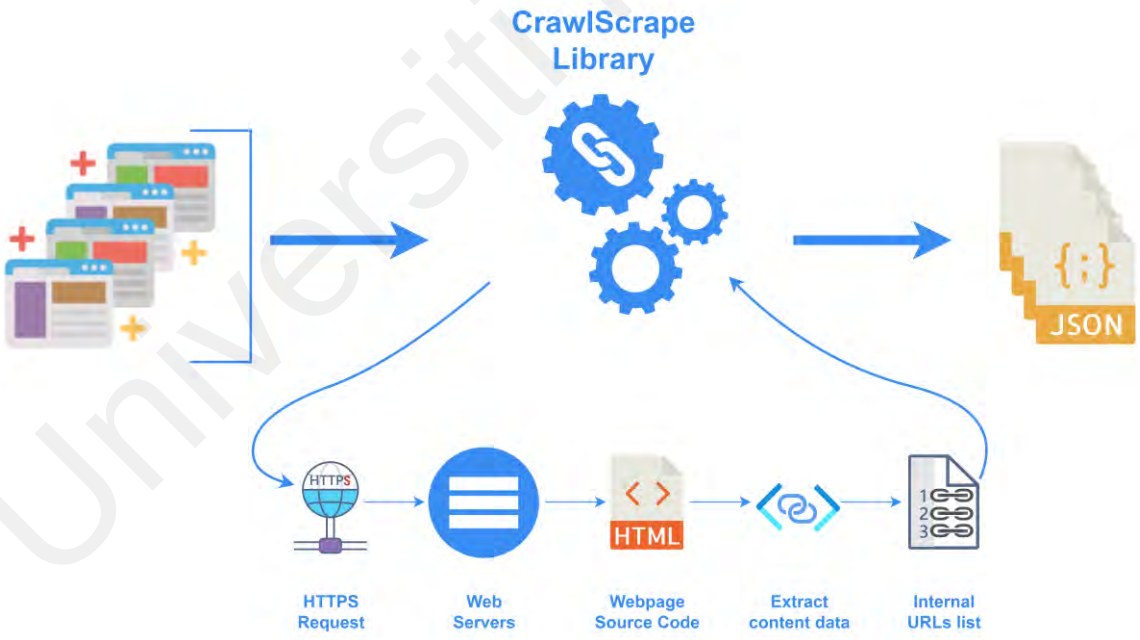

Figure 4.10: Architecture and main components of the CrawlScrape library

Algorithm II states the mechanism for web content extraction used in the CrawlScrape library.

Algorithm II: Crawling and scraping
Import collected websites For each website Crawl URL links Identify internal, external, documents, and executable links Store internal links For each internal link Retrieve HTML code Parse HTML code Retrieve textual contents from HTML tags Retrieve visual source links from HTML tags Extract given features Store textual contents, visual content links, and extracted features Store website content in JSON format Collect and export all JSON files for all websites

The result of the CrawlScrape library of each website in the provided websites list is as Table 4.4 demonstrates.

Table 4.4: CrawlScrape output for each website

File	Details
Metadata	The library produces a metadata JSON file for each website in the links provided. The name of this file will be metadata
Internal URLs details	The library creates a detailed JSON formatted file for each crawled internal URL of a specific website. The name of this file will be the URL of the internal webpage

Algorithm III illustrates the web content extraction algorithm used in the CrawlScrape library.

Algorithm III: Extracting web content data

```
Import collected websites
For each website
    Crawl URL links
    Identify internal and external links
    Store internal links
    For each internal link
        Retrieve HTML code
        Parse HTML code
        Retrieve textual contents from HTML tags
       Store all textual contents
   Store website content in JSON format
Merge and export all JSON files for all collected websites
```

4.4.3. Datasets Description

This section describes the datasets used in the experiment. This study investigates differences in the performance of topic models on conventional document/article data and webpage content. As aforementioned, this study evaluates the benchmark topic models, the proposed HTM topic model, and the cyber parental control framework using three datasets. The following subsections introduce each dataset and describe its data.

a) ***Dataset I:*** *Conventional document-based data*. The study's first dataset comprises textual data from various documents and articles, largely sourced from Wikipedia due to its wide topic range and freely accessible nature (Fu et al., 2021; Ma et al., 2020; O'callaghan et al., 2015; Röder et al., 2015; Syed & Spruit, 2017; Wang et al., 2020). The study refers to data derived from online page content as Conventional Data-based (CD-based). The complete dataset, containing Wikitext source and embedded metadata, is downloadable as a single XML file from Wikipedia (https://dumps.wikimedia.org/enwiki/latest/). The data used in this study, drawn on

22-Feb-2023, consists of more than 7 million articles from a multitude of categories, with a file size of over 20 Gigabytes. A subset of 50,000 randomly selected articles was used for Experiment II (as shown in Section 5.3). This subset contains 55,775,941 words, including around 584,132 unique words, with an average of 1,115 words per article.

b) ***Dataset II****: Web content-based data*. The study refers to data derived from online page content as Web Content-based (WC-based) data, which includes text, images, videos, and other media. However, this dataset only utilises textual content to evaluate the benchmark topic models and the proposed HTM topic model. The methods of data collection, extraction, and preprocessing of web pages are detailed in Section 4.4.1. The study obtained these websites from DMOZ and Alexa, with the dataset representing categories like arts, business, computers, games, health, news, science, society, sports, and kids & and teens. A total of 125,000 web pages were randomly selected for this dataset, containing around 55,753,919 words, of which approximately V=400,230 were unique, with an average of 446 words per webpage.

c) ***Dataset III****: Ground truth data*. A ground truth dataset is a set of data labelled as the "true" and "false" values for a certain field. This dataset contains about 2 million web pages collected from about 7,000 websites, evenly split with 3,500 each of objectionable and unobjectionable websites. A full description of creating the ground truth dataset is explained in the subsequent section.

4.4.4. Ground Truth

The ground truthing concept is used in machine learning as well as in other domains. It serves as a point of reference or a benchmark for determining how other data or models compare to the one being examined. In fact, creating a ground-truth dataset requires a careful inspection and annotation of the data, which is a time-consuming and labour-

intensive process. Nevertheless, a high-quality ground-truth dataset is significantly valuable for training and evaluating machine learning models.

In this study, a ground truth dataset consists of web pages that have been manually labelled with the objectionability topics contained in them. The significance of the ground truth dataset is that the machine learning models can be trained and tested to determine how accurately they are able to identify and classify the objectionability topics these web pages contain. A reliable ground truth dataset is essential for building effective cyber parental control models and verifying the validity of new detection methods.

As aforementioned, there is a lack of a standard dataset in the current web content filtering studies, as shown in Table 2.6. Moreover, no ground truth exists for objectionable and unobjectionable websites publicly available for cyber parental control research. The lack of publicly accessible datasets with a reliable ground truth has prevented in the past a fair and coherent comparison of different methods proposed in the field of cyber parental control. Consequently, there is a significant need for ground-truthing of objectionable and unobjectionable websites for cyber parental control. Thus, this study creates an objectionable ground truth dataset.

Creating the objectionable ground truth dataset includes the following steps:

a) *Specify the task*. In the first step, this study defines the criteria of objectionable web content topics (as shown in Section 2.1.1) and collects web pages based on them.
b) *Data collection*. The second step is to collect the data for building the ground-truth dataset. This study collects web pages from data from various sources, as Section 4.4.1 addressed.
c) *Data extraction*. Since the collected data are web pages containing HyperText Markup Language (HTML), an additional step is required to extract their textual

contents. This study designed and developed a crawling and scraping tool to achieve this step (Section 4.4.1).

d) *Data pre-processing*. After extracting the textual content data in the previous step, this step pre-processes the data to be ready for labelling. The pre-process includes a few steps, which Section 4.4.1 addressed.

e) *Data labelling*. This step aims to label the collected and extracted web content data based on their source categorization, features classification, and extracted topic classification. The extracted topic will be classified as either objectionable or unobjectionable. As aforementioned, this study conceptualizes objectionable web content terms as textual content that children users oppose on the web, including, but not limited to, pornography, violence, drugs, hate, racism, sexual, homicide, gambling, and weapons. This step labels the content of web pages based on this definition as objectionable and unobjectionable.

f) *Validate*. The final step of creating the ground truth dataset is validating the labels and inspecting the bias of the data. This step requires investigating to what extent the source labelling and the human manual labelling agree. The study's first experiment describes the validation of the created ground truth in detail (as shown in Section 5.2)

4.4.4.1. Data Description

The ground truth dataset contains raw data (in a JSON format) of objectionable and unobjectionable websites. The ground truth dataset contains two files, an objectionable dataset file and an unobjectionable dataset file. Each file contains the exact number of attributes. This research selects these attributes based on similar previous datasets (Singh, 2020; Vrbančič et al., 2020). Most of these attributes were extracted with the help of Selenium (Selenium for Python, 2021) and BeautifulSoup libraries.

The first file, the domain metadata file, is named *metadata.json*. This file gives an overview of the websites and their features. Table 4.5 details the attribute name, data type, and description of each field of this file.

Table 4.5: Description of the attributes of all websites of the objectionable ground truth dataset

Attribute	Data type	Description
domain	String	A code (D#) replacing the domain name of the website
geo_locs	String	Names of the countries based on the 'domain's IP Address location using GeoIP Databases ("GeoIP Database,")
domain_length	Numeric	Number of domain characters
tld	String	TLD of the webpage using tld Library ("Tld Library,")
avg_time_response	Numeric	The response time of a webpage request in milliseconds
start_scrapping_timestamp	Numeric	The timestamp in milliseconds of scrapping the webpage
domain_tls_ssl_certificate	Numeric	0 if the webpage does not use a certificate 1 if the webpage uses s certificate
internal_urls_no	String list	Number of web pages that have been collected from the website
internal_urls	Numeric	List of all internal URLs that have been collected from the website
source	String	The collected source of the website
label	String	A categorical string of the webpage, either objectionable or unobjectionable

The second file, the internal web pages detailed file, is named web pages_detail.json. This file gives detailed information on each collected website's web pages (internal URLs) and features. Table 4.6 details the attribute name, data type, and description of each field of this file.

Table 4.6: Description of the attributes of all web pages (URLs) of the objectionable ground truth dataset

Attribute	Data type	Description
url	String	A code (D#_URL#) replacing the URL of the webpage
domain_name	String	The code (D#) of the domain that the webpage belongs to
created_time	String	Time created the record (format yyyy-MM-dd HH:mm:ss)

geo_loc	String	Name of the country based on the 'webpage's IP Address location using GeoIP Databases ("GeoIP Database,")
domain_length	Numeric	Number of domain characters
url_length	Numeric	Number of URL characters
time_response	Numeric	The response time of a webpage request in milliseconds
html_char_length	Numeric	Number of characters in the full HTML
text_char_length	Numeric	Number of characters in all visible texts
textual_tags_cnt	Numeric	Number of the list of all visible texts on the webpage
visual_content_no	Numeric	Number of the list of all visuals on the webpage
label	String	A categorical string of the webpage, either objectionable or unobjectionable
label_details	String	A sub-categorical string of the webpage, including but not limited to porn, gambling, erotica, sport, news, and kids
tld	String	TLD of the webpage using Tld Library ("Tld Library,")
protocol	String	Name of the protocol used by the webpage URL (HTTP, HTTPS, and FTP)
tls_ssl_certificate	Numeric	False if the webpage does not use a certificate True if the webpage uses s certificate
source	String	The collected source of the website

4.5. Summary

This chapter presents the core components of the cyber parental control framework and the classification model. The core of the framework is the proposed HTML Topic model (HTM). The chapter begins by describing the formulation and notation of the proposed HTM topic model. It then explains the generative process and the mathematical model of the HTM topic model.

The chapter then describes the design of the classification framework and its three main layers. The first two layers incorporate whitelisting and blacklisting, which is a lightweight approach to classifying web pages. If the webpage is not classified on either layer, the webpage then enters the final layer of the framework involving the HTM topic model. The final layer of the framework includes three modules, each of which is responsible for carrying out the necessary steps. The first module scrapes and extracts the

textual content of the target webpage. The result of this module is then input to the topic modeling module, which employs the proposed HTM topic model to interpret topics in the webpage. The final module uses trained classifier models in order to classify the webpage into objectionable or unobjectionable categories. Upon completion, both whitelist and blacklist databases will be updated. The chapter then addressed the operational characteristics of the proposed cyber parental control framework. The chapter concludes by elaborating on the datasets used to evaluate the benchmark topic models, the proposed HTM model, and the classification framework in the following chapter.

CHAPTER 5: EVALUATION OF CYBER PARENTAL FRAMEWORK

The key characteristics of the framework, described in Chapter 4, lie in the final layer, where the HTM topic model extracts coherent topics from web data. A thorough evaluation is essential to demonstrate the coherency of the HTM model and the effectiveness and accuracy of the framework. This chapter describes four extensive experiments conducted in sequence to evaluate the cyber parental control framework. In addition, the HTM topic model is benchmarked against topic models widely used in literature to demonstrate the improvements achieved in classifying objectionable web content.

The series of experiments includes objectionable dataset ground truthing, topic models benchmark performance on web content data, HTM topic model performance on web content data, and web content classification accuracy. Figure 5.1 shows these series and their order to achieve the aim of this study. The following sections describe each experiment as follows:

a) *Experiment I*. This experiment investigates the reliability of the created ground truth using the kappa coefficient score.
b) *Experiment II*. This experiment uses two different sources of data to demonstrate the performance drawback of the benchmark topic models on web content data.
c) *Experiment III*. This experiment evaluates the proposed HTM topic model against the best performance model of the benchmark topic models when performing on web content data. It uses the same dataset source as the second experiment.
d) *Experiment IV*. This experiment evaluates the proposed cyber parental control framework with a real ground truth dataset containing many web pages' content data.

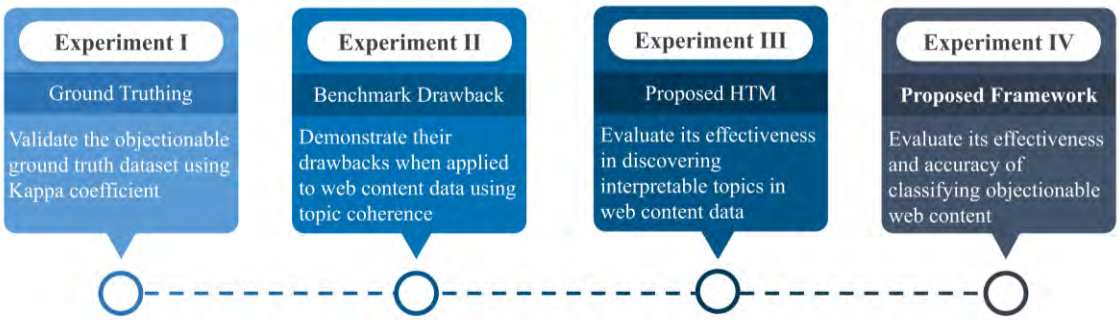

Figure 5.1: Experimental methodology of this study

5.1. Evaluation Measurements

Various evaluation measurements can be utilized to evaluate the performance of a machine learning model. The following series of experiments evaluate the existing topic modeling, the HTM topic model, and the cyber parental control framework using the following:

a) *Evaluating Topic Model*. This study evaluates the benchmark topic models and the proposed HTM topic models using topic coherence metrics, including C_V, C_{UMass}, C_{UCI}, and C_{NPMI} metrics, as Section 3.4 explained. These metrics evaluate the quality of topics generated by the benchmark and the HTM topic models and have proven their alignment with human judgment (Röder et al., 2015).

b) *Evaluating Classification Framework*. This study uses accuracy, precision, recall, and F1 scores to evaluate the cyber parental control framework. These metrics indicate the framework's predictivity and rely on TP, TN, FP, and FN calculations, as Section 2.3.3 explained. Assumed N_{unobj} is the number of unobjectionable web pages, and N_{obj} is the number of objectionable web pages, then the parameters for computing the metrics are as follows:

- $n_{obj \to obj} = TP$: Number of objectionable URLs correctly classified as objectionable.
- $n_{unobj \to unobj} = TN$: Number of unobjectionable URLs correctly classified as unobjectionable.

- $n_{unobj \rightarrow obj} = FP$: Number of unobjectionable URLs which are classified as objectionable.
- $n_{obj \rightarrow unobj} = FN$: Number of objectionable URLs which are classified as unobjectionable.

5.2. Experiment I: Ground Truth Dataset

The final step of creating the objectionable ground truth dataset is validating the labels and inspecting the data's bias and mislabeling. This step requires investigating to what extent the source labelling and the human manual labelling agree. This experiment aims to ensure the correctness of labels in the created ground truth dataset, as incorrect labels can lead to inaccurate conclusions and poor model performance.

5.2.1. Experiment Aims and Description

This experiment validates the accuracy of labelling the objectionable ground truth dataset using inter-rater agreement. This study uses the Kappa coefficient score to calculate the agreement of the labels in the objectionable dataset. Kappa Cohen's coefficient is "a statistical measure of inter-rater reliability or agreement used to assess qualitative documents and determine the agreement between two raters".

The Kappa coefficient measures the agreement between ground truth labelling and manual human labelling. In order to calculate that, 1600 websites were chosen randomly, representing 20% of the total number of websites in the dataset, and labelled manually as objectionable and unobjectionable. Five people experienced in content classification and categorization were selected for this task. In this way, the study aimed to demonstrate the presence of selection bias in any of the sources using the Kappa coefficient. The Kappa coefficient was applied to compare the manual labels of the randomly selected 1600 websites with the original labels from the source (each source has its categorization). The

following equations were used to calculate the agreements of the manual and source labels:

$$observed\ agreement = A + D \tag{1}$$

$$expected\ agreement = \frac{\big((A+B)\times(A+C)\big) + \big((C+D)\times(B+D)\big)}{n} \tag{2}$$

$$kappa = \frac{observed\ agreement - expected\ agreement}{n - expected\ agreement} \tag{3}$$

Where A is the number of agreements on the first label, B is the number of no agreements on the first label, C is the number of no agreements on the second label, D is the number of agreements on the second label, and n is the number of dataset records. The interpretation of the Kappa score is as Table 5.1 tabulates.

Table 5.1: Kappa score interpretation

Kappa Score	Interpretation
< 0	No agreement
$0.0 - 0.20$	Slight agreement
$0.21 - 0.40$	Fair agreement
$0.41 - 0.60$	Moderate agreement
$0.61 - 0.80$	Substantial agreement
$0.81 - 1.00$	Almost perfect agreement

5.2.2. Result And Discussion

The agreements of the manual and source labels for the randomly selected websites are calculated using the Kappa coefficient. The Kappa coefficient comparing the human (manual) and source (automatic) labelling of 20% of the websites in the ground truth dataset was 0.87 (calculations in Table 5.2), indicating very high agreement and, thus, low selection bias. Table 5.2 shows the labelling results of these 1600 websites by both labelling sources.

Table 5.2: Result of both ground truth labelling and manual human labelling

Source (automatic) classification	Human (Manual) Classification		
	Objectionable	**Unobjectionable**	**Subtotal**
Objectionable	730	70	800
Unobjectionable	10	790	800
Subtotal	740	860	1600

The calculation of Kappa's score is shown below:

$$Observed\ agreement\ =\ 1520$$

$$Expected\ agreement\ =\ ((800\ \times\ 740)\ +\ (800\ \times\ 790))/1600\ =\ 765$$

$$Kappa\ score\ =\ (1520\ -\ 765)/(1600\ -\ 765)\ =\ 0.904$$

$$Kappa\ score\ >\ 0.904$$

The Kappa score indicates an almost perfect agreement between human and ground truth classification. Therefore, the dataset can be considered a ground truth dataset for objectionable and unobjectionable web content data. For this reason, this study uses this dataset to evaluate the proposed cyber parental control framework in experiment IV (Section 5.5).

5.3. Experiment II: Existing Topic Models

The second experiment of this study evaluates the benchmark topic models and their performance. Since this study focuses on web content data, the benchmark models' evaluation compares these models' performance on conventional and web content data. This experiment demonstrates the limitations of the existing topic models when applied to web content data. Therefore, there is an essential need for a web content topic model that effectively learns interpretable topics in web content data.

5.3.1. Experiment Description

This experiment evaluates the benchmark topic models (CTM, DMR, HLDA, LDA, PTM, and sLDA) and uses *Python* programming language with the help of Gensim (Rehurek, 2011) and Tomotopy (bab2min) libraries to implement these models. The main parameters used to build these topic models are the corpus in a BoW format, id2word mapping to the dictionary, and the number of topics. Other parameters such as alpha, random state, chunk size, and passes are set to default values.

In order to evaluate these models based on their performance on two sources of data (conventional document data and web content data), this experiment utilized *Dataset I* (as shown in Subsection A in Section 4.4.3) and *Dataset II* (as shown in Subsection B in Section 4.4.3). This experiment is evaluated by topic coherence using C_{UMass}, C_{UCI}, C_{NPMI}, and C_V metrics. Using these metrics, the performance of the models for each dataset and given the number of topics $K \in \{1 \dots, 100\}$ will then be compared in the following subsections. The following section discusses the results of all models' comparisons on both datasets using each metric.

5.3.2. Experiment Result

The following subsections present each metric evaluation and comparison of the benchmark topic models on both datasets. The results are discussed and visualized using a line chart. The x-axis of the chart represents the number of topics, and the y-axis represents the coherence metric score. Each benchmark model is plotted in a different line colour, which is explained in the legend of each graph. These results are then discussed in the following section (as shown in Section 5.3.3).

a) *Coherence UMass Evaluation*. Figure 5.2 and Figure 5.3 show each model's C_{UMass} topic coherence score on CD-based and WC-based datasets, given the different numbers of topics. The overview of both figures shows a significant drop in the

benchmark topic models' performance when applied to web content data. Figure 6 shows that not only did all models perform significantly worse on the WC-based dataset compared to the CD-based dataset across the number of topics, but the difference in performance between the WC-based dataset and the CD-based dataset is notably large. The CTM model consistently achieved the highest topic coherence score given different numbers of topics for CD-based datasets, whilst the LDA model scored the lowest. The vast difference in results on both datasets indicates that these models failed to capture the coherence of topics on webpage content data. Intriguingly, Figure 6 reveals that while the LDA model performed the worst for CD-based datasets, it performed the best for WC-based datasets, achieving the highest topic coherence score compared to other models. The difference in the LDA performance on both datasets is slight, which indicates that the LDA model is the most stable model among the benchmark topic models based on the C_{UMass} metric.

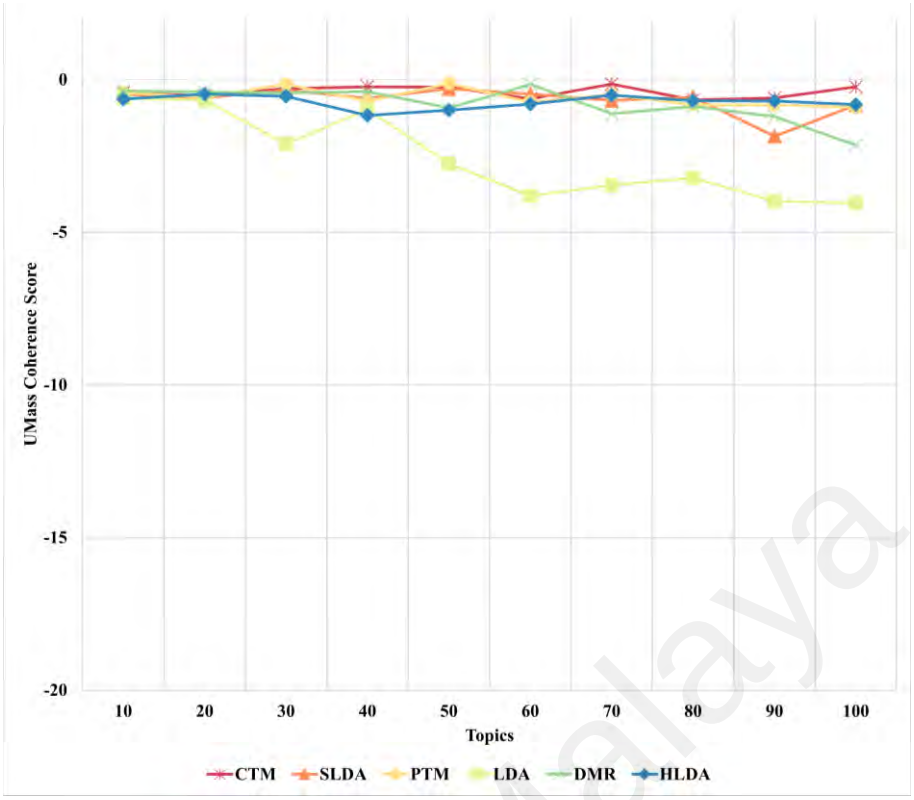

Figure 5.2: C_{UMass} coherence score of the benchmark topic models when applied on CD-based dataset

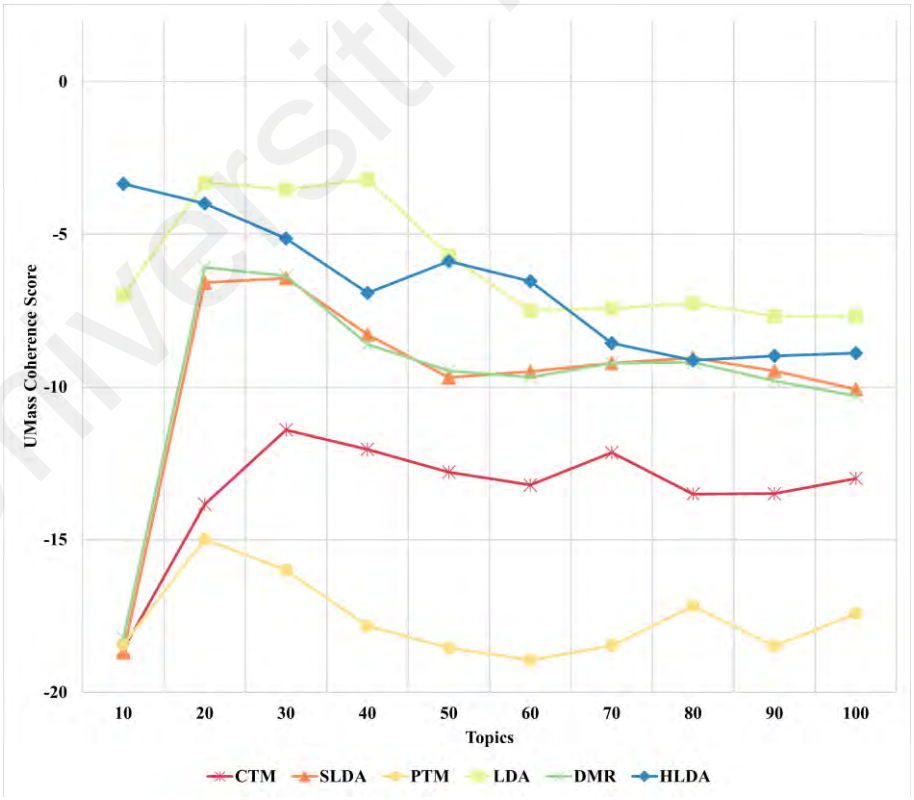

Figure 5.3: C_{UMass} coherence score of the benchmark topic models when applied on WC-based dataset

b) *Coherence NPMI Evaluation*. Figure 5.4 and Figure 5.5 show each model's C_{NPMI} topic coherence score on CD-based and WC-based datasets, given the different numbers of topics. The overall performance of the benchmark topic models is significantly worse in the WC-based dataset compared to the CD-based dataset, and the difference in performance between both datasets is vast. The sLDA and DMR models achieved the highest topic coherence scores, given different numbers of topics for CD-based datasets, whilst the PTM model scored the lowest. The considerable difference in results on both datasets indicates that these models failed to capture the coherence of topics on webpage content data. Figure 8 also shows that the drop in the LDA model performance was the lowest among the benchmark topic models based on the NPMI coherence metric. This slight drop in performance indicates that the LDA model is the most stable model among the benchmark topic models based on the C_{NPMI} metric. When scrutinizing the NPMI scores of the topic models, it is noticeable that they keep decreasing as the value of K increases when applied to the CD-based dataset, unlike their steady performance on the WC-based dataset despite the increase in K.

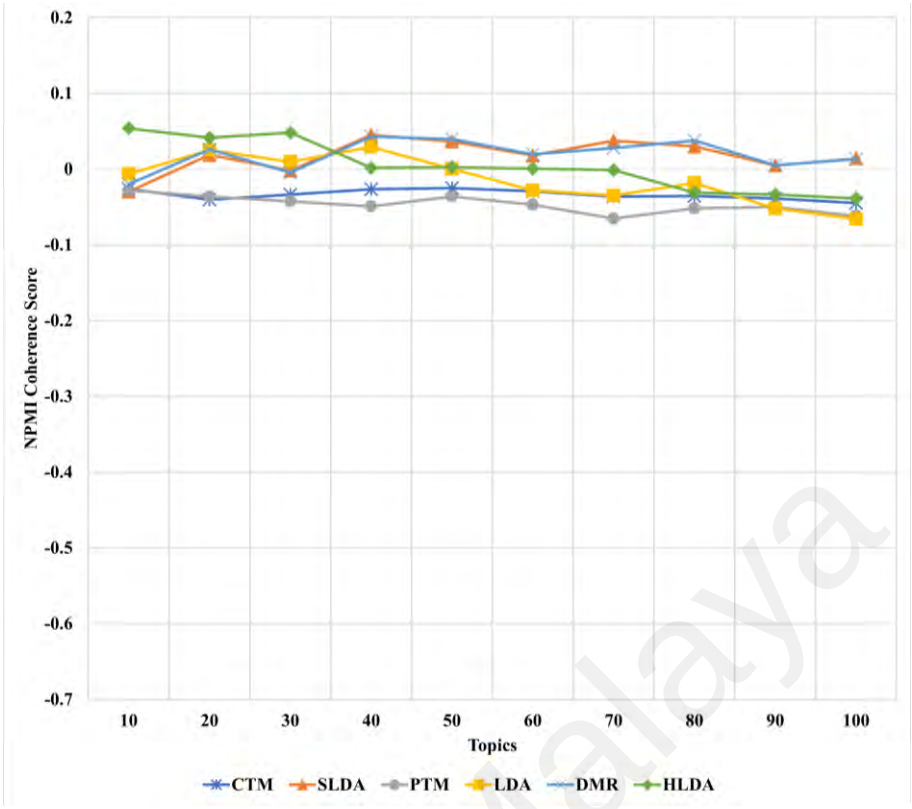

Figure 5.4: C_{NPMI} coherence scores of the benchmark topic models when applied on CD-based dataset

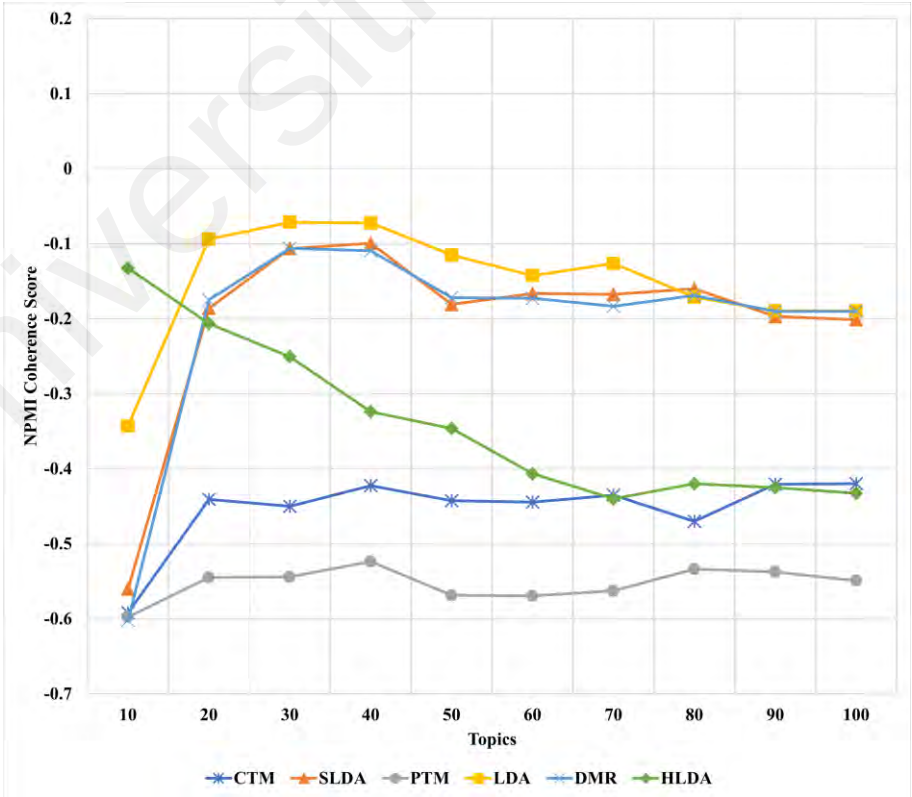

Figure 5.5: C_{NPMI} coherence scores of the benchmark topic models when applied on WC-based dataset

c) *Coherence V Evaluation*. Figure 5.6 and Figure 5.7 show topic modes' C_V coherence scores on CD-based and WC-based datasets. Based on the figures, all benchmark models perform better on the WC-based dataset than on the CD-based dataset. However, the difference in performance is not marginal. Although they perform better on WC-based datasets than CD-based datasets, the increase in topic coherence score is modest, indicating only a slight enhancement. The figures show that the DMR and sLDA models performed best on both CD-based and WC-based datasets. Meanwhile, the LDA model performed worst for both CD-based and WC-based datasets. Interestingly, the figures reveal that whilst the LDA model performed the worst, it showed a sharp rise in performance from 0-10 topics before dipping and stagnating after that as the number of topics increased. The difference in the LDA model's performance was the lowest, which indicates that the LDA model is the most stable model among the benchmark topic models based on the C_V metric.

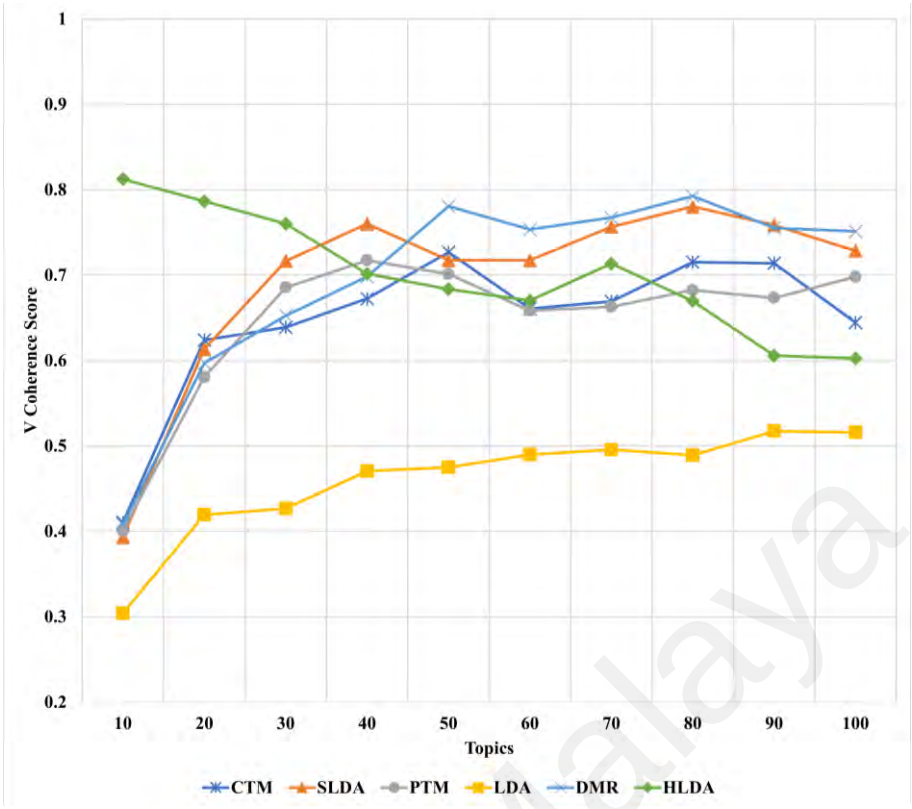

Figure 5.6: Cv coherence scores of the benchmark topic models when applied on CD-based dataset

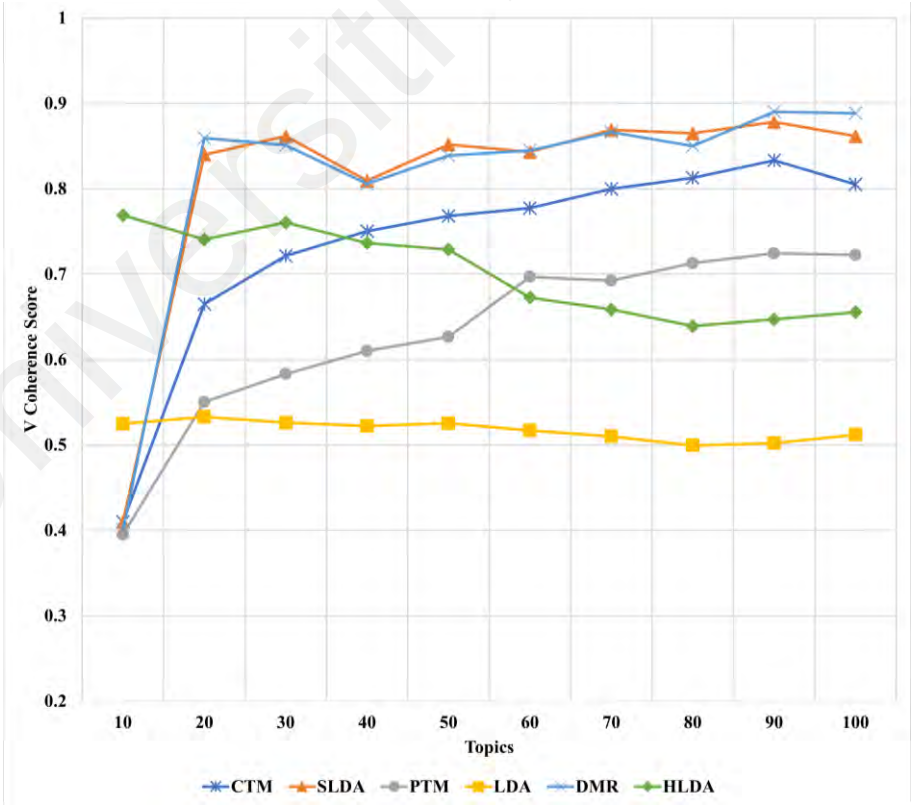

Figure 5.7: Cv coherence scores of the benchmark topic models when applied on WC-based dataset

d) *Coherence UCI Evaluation*. Figure 5.8 and Figure 5.9 show each model's C_{UCI} coherence score on CD-based and WC-based datasets, given the different numbers of topics. The overall performance of the topic models is significantly worse in the WC-based dataset compared to the CD-based dataset, and the difference in performance between both datasets is vast. The sLDA, DMR, and CTM models consistently achieved the highest topic coherence scores given different numbers of topics for CD-based datasets, while the LDA model scored the lowest. The performance of the CTM model on web content data witnesses the largest failure among the benchmark topic models, with a drop of 600% based on the CUCI metric. The large difference in results on both datasets indicates that these models failed to capture the coherence of topics on webpage content data. Scrutinizing Figure 7 helps to discover an interesting observation regarding the LDA model performance on both datasets. The LDA model performed the worst for CD-based datasets. It, however, performed the best for WC-based datasets, achieving the highest topic coherence score in comparison to other models. The difference in the LDA performance on both datasets is slight, which indicates that the LDA model is the most stable model among the benchmark topic models based on the C_{UMass} metric.

Figure 5.8: C_{UCI} coherence scores of the benchmark topic models when applied on CD-based dataset

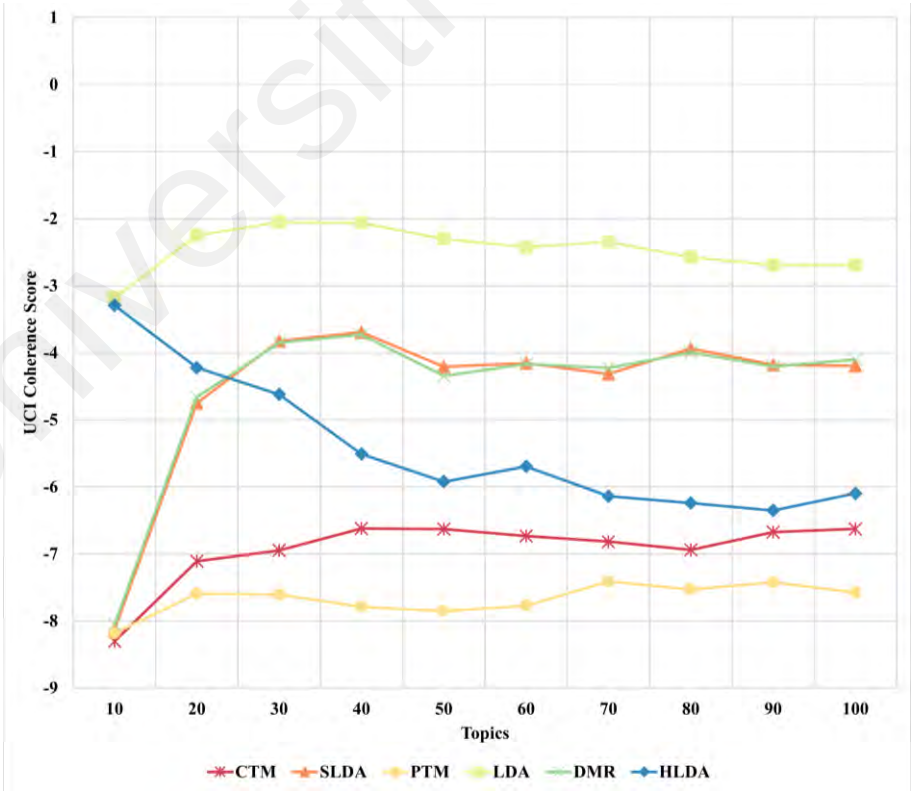

Figure 5.9: C_{UCI} coherence scores of the benchmark topic models when applied on WC-based dataset

5.3.3. Experiment Discussion

The abovementioned comparison reveals a few indications and observations. The overall performance of these topic models on conventional document data outweighs their performance on documents with web content-based data. The difference in performance was significant for some topic models, such as the PTM and CTM, and it was inconsiderable for other models, such as the LDA. The C_{UCI} and C_{NPMI} appear to prefer fewer topics on conventional document data.

In contrast, the C_V topic coherence favours a higher number of topics for all models except the HLDA. It is worth mentioning that the sLDA scores are similar to the DMR scores given all metrics and in both datasets, which is likely due to their methodological similarities (Mimno & McCallum, 2008). Although the authors (Mimno & McCallum, 2008) emphasize the difference between their proposed DMR model and the sLDA model, both models perform comparably similarly in our study.

Scrutinizing the findings of the comparison, a few interesting observations are discovered. The difference in the LDA performance on both datasets is slight, which indicates that the LDA model is the most stable model among the benchmark topic models based on all coherence metrics. Another observation is that there is no dominant topic model among the benchmark topic models based on all metrics and for both datasets. However, it is noticeable that the LDA outperforms other models in many ways by appearing as the top score model per the CUMass, CUCI, and CNPMI metrics on the webpage content data. Models with the lowest performance on the webpage contents data were HLDA, CTM, and PTM in our experiment. Given these factors, there is a need for an enhanced model designed for web content-based data.

5.4. Experiment III: HTML Topic Model

As aforementioned, the goal of the proposed HTM topic model is to improve the topic modeling of web content data. The third experiment evaluates the HTM topic model against the LDA topic model, which performed the best on web content data among the benchmark topic models, as shown in the previous experiment (as shown in Section 5.3.3). This experiment demonstrates the effectiveness of the HTM model in discovering interpretable topics and term patterns of web content data compared to the existing topic models. The following sections describe the experiment, present the results, and discuss these results.

5.4.1. Experiment Aims and Description

This experiment evaluates the HTM topic model against the LDA model. *Python* programming language with the help of Gensim (Rehurek, 2011) and NLTK (Bird et al., 2009) libraries are used to implement these models. Similar to the previous experiment, the main parameters used to build these topic models are the corpus in a BoW format, id2word mapping to the dictionary, and the number of topics. Other parameters such as alpha, random state, chunk size, and passes are set to default values.

This experiment utilized *Dataset II* (as shown in Section 4.4.3) to evaluate both models based on their performance in web content data, using C_{UMass}, C_{UCI}, C_{NPMI}, and C_V metrics. The subsequent section then compares the performance of the HTM and LDA topic models for the web content data, given the number of topics $K \in \{1 \dots, 100\}$. The following section discusses the results of both models and the indication of these results.

5.4.2. Experiment Result

The following subsections present each metric evaluation and comparison of the HTM and LDA topic models on the web content-based dataset. Similar to the previous experiment, the results are discussed and visualized using a line chart. The x-axis of the

chart represents the number of topics, and the y-axis represents the coherence metric score. Each model is plotted in a different line colour, which is explained in the legend of each graph. These results are then discussed in the following section (as shown in Section 5.4.3).

a) *Coherence UMass Evaluation*. Figure 5.10 addresses the comparison results of the LDA and HTM topic models per the C_{UMass} metric, showing the HTM topic model performing superiorly. The C_{UMass} value of the HTM model started at about -1.5 and remained steady over the number of topics. In contrast, the LDA model's performance declined as the number of topics increased. The value of the LDA model started at -7 and increased to more than -4 when the number of topics was 10, and then dropped to slightly above -10 at 25 topics. These results indicate that the HTM topic model performs better for the web content data. This is because the UMass is an intrinsic measure that considers preceding and succeeding terms in the list. This feature allows the UMass metric to relate topics within each HTML tag and among tags on each webpage. This interprets the significant improvement of the HTM topic model compared to the LDA topic model based on this metric.

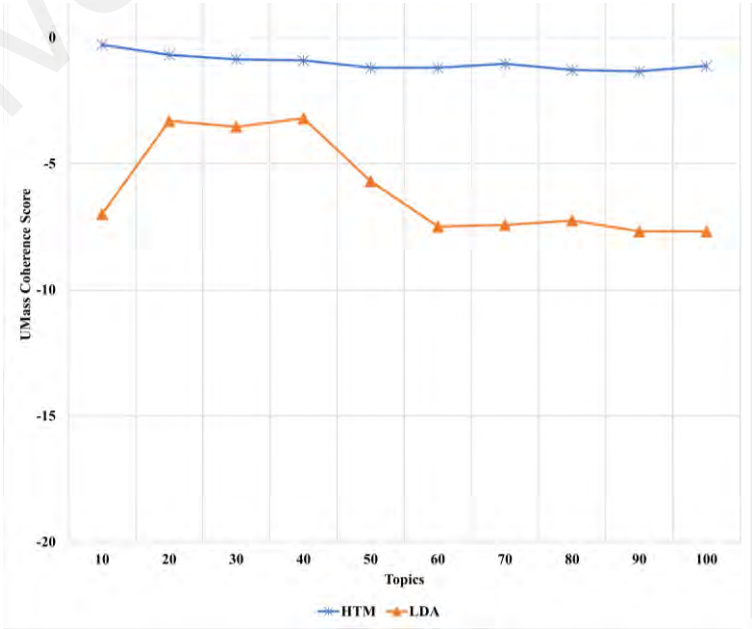

Figure 5.10: C_{UMass} coherence scores of the LDA and HTM topic models

b) *Coherence NPMI Evaluation*. Figure 5.11 compares the LDA and HTM topic models per the C_{NPMI} metric. The results suggest that the HTM model performs slightly better than the LDA model when the number of topics is high. Like the C_{UCI} metric, the HTM model performs significantly better than the LDA model with a small number of topics ($K < 5$). The difference then decreases to less than 0.5 units between the two models. The C_{NPMI} value of the HTM model fluctuated around -0.15 when the topic number exceeded 10, and the value of the LDA model remained steady at slightly more than -0.2 when the topic number exceeded 22. This insignificant difference indicates that the performance of the HTM model slightly surpassed the LDA model for the web content data. The comparison results of the C_{NPMI} coherence score are similar to the C_{UCI} coherence scores, which is expected since both metrics rely on the pointwise function.

Figure 5.11: C_{NPMI} coherence score of the LDA and HTM topic models

c) *Coherence V Evaluation*. Figure 5.12 illustrates the results of this metric for the LDA and HTM models. The overall results of this metric suggest that the HTM topic model

outweighs the LDA model for any number of topics. The HTM topic model scored a C_V value of ≥ 0.9 when the number of topics exceeded 4. The LDA C_V value started at less than 0.3 and significantly increased to reach more than 0.7 when the topic number was 10. The value then decreased to less than 0.7. In comparison, the HTM topic model is superior to the LDA model, given the C_V metric, for modeling the topics of web content data. Literature indicated that maximizing C_V value enhances human interpretability (Röder et al., 2015). Therefore, this result means that the HTM model learns better web content interpretable topics than the LDA model. This result is because each webpage tag contains related terms and topics, which the HTM considers in its design. Considering this allows the HTM model to learn the indirect coherence between these terms of each tag, thus enhancing its C_V score.

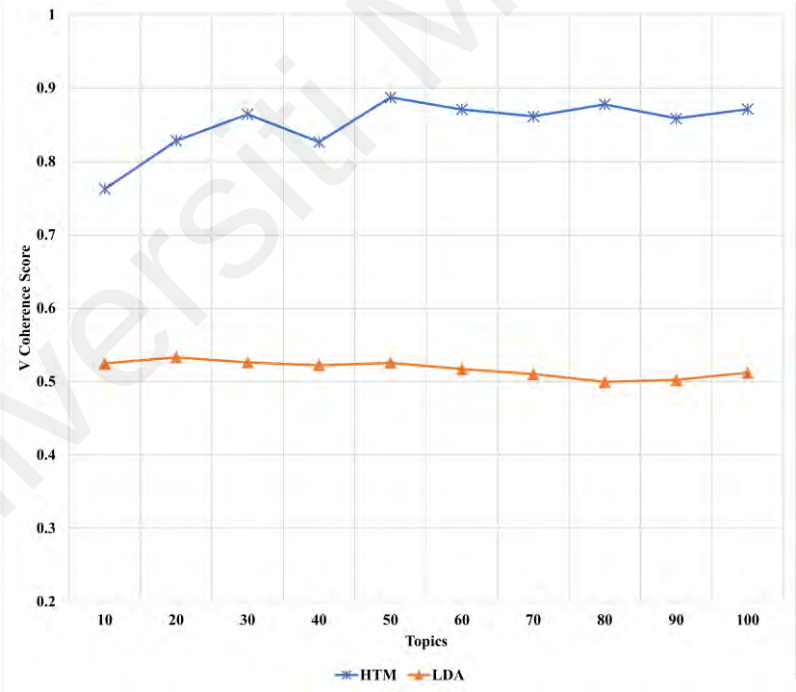

Figure 5.12: C_V coherence score of the LDA and HTM topic models

d) *Coherence UCI Evaluation*. Figure 5.13 illustrates the results of the C_{UCI} for both the LDA and HTM topic models. The results suggest that the LDA model is slightly better than the HTM topic model when the number of topics is high. However, when scrutinizing the C_{UCI} value of the model, it is noticeable that the HTM model performs

significantly better than the LDA model with a small number of topics (K < 5). The difference then decreases to less than 1 unit between the two models. The C_{UCI} value of the HTM model fluctuated between -8 and -9 when the topic number exceeded 13, and the value of the LDA model remained steady at less than -8 when the topic number exceeded 20. This insignificant difference indicates that the performance of both models is similar, with a slight surpass for the LDA model over the HTM model for the web content data. The nature of the UCI metric causes the similarity of both models' results, and the UCI does not rely on the given corpus of the WC-based dataset. In this case, considering the HTML tags did not significantly improve the LDA model, and therefore, both models resulted in similar C_{UCI} scores.

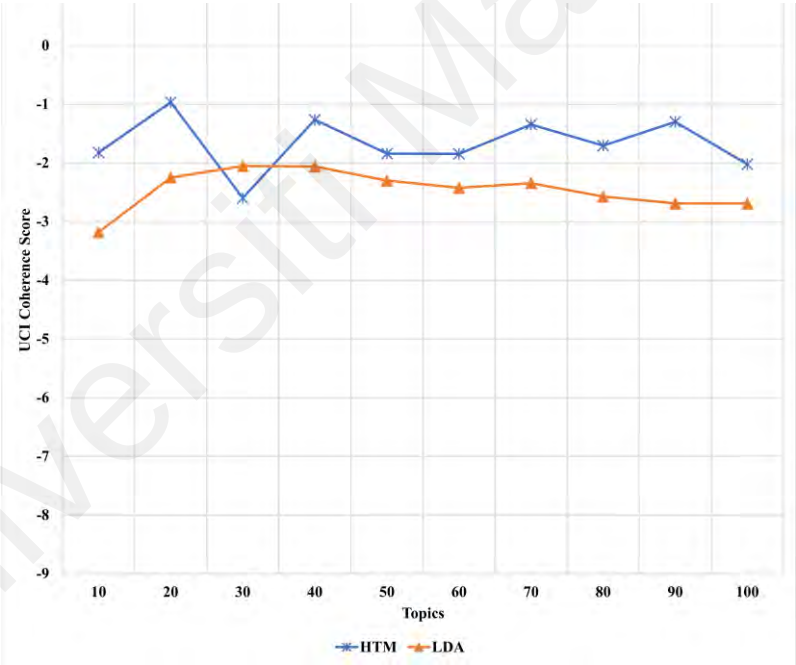

Figure 5.13: C_{UCI} coherence score of the LDA and HTM topic models

5.4.3. Experiment Discussion

The abovementioned results offer several indicators and findings. The overall performance of the HTM model is superior to the performance of the LDA model. The performance difference was substantial for some metrics, such as C_{UMass}, C_V, and C_{NPMI} and negligible for C_{UCI}. The results also show that the HTM model converges when topics

exceed 10 whereas the LDA converges when topics exceed 20-30. The fast convergence of the HTM model is a significant benefit that can be highly valued in certain application settings where time resources are limited (Altarturi, 2022).

Scrutinizing the findings of the comparison, a few interesting observations are discovered. Among the four measurements, three metrics show an improvement in the HTM model, and only one metric shows an insignificant drawback of the HTM model. The C_{UMass} coherence score shows that the HTM model performs significantly better than the LDA model, with a steady value over the number of topics, while the LDA model's value decreases as the number of topics increases. The improvement of the HTM model was slightly more than 89% of the LDA using the CUMass metric. This improvement is due to the UMass metric, which considers preceding and succeeding terms in the list and allows the HTM model to relate topics within each HTML tag and among tags on each webpage.

Similarly, the C_V coherence score shows that the HTM model outperforms the LDA model for any number of topics, with a C_V value of ≥ 0.9 when the number of topics exceeds 4. The improvement of the HTM model was slightly more than 36% of the LDA using the C_V metric. These results indicate that the HTM model learns better web content topics than the LDA model, which may enhance human interpretability, as some previous studies argued (D. Mimno et al., 2011; Newman, Lau, Grieser, & Baldwin, 2010; Röder et al., 2015). These results indicate that considering HTML tags when applying topic modeling on web content data increases the quality of generated topics, which answers the third research question of this study.

The C_{UCI} and C_{NPMI} coherence scores show that the HTM model performs slightly better than the LDA model when the number of topics is high. However, when the number of topics is low, the HTM model significantly outperforms the LDA model. Another

improvement of the HTM model was recorded using the C_{NPMI} metric with more than 26%. This improvement is due to the fact that the HTM model considers the indirect coherence between related terms of each tag, which enhances its performance in learning coherence topics.

The results show that the enhancement made by the HTM model was vast based on C_{UMass} and C_V metrics yet slight based on C_{UCI} and C_{NPMI} metrics. These phenomena suggest that the HTM model made a significant enhancement of the LDA model in generating topics that are semantically related and coherent in terms of the overall corpus. However, the HTM model was similar to the LDA model in terms of generating a strong association within each topic. This result is due to several reasons related to the nature of each coherence metric.

C_{UMass} coherence measures the degree of semantic coherence between the words in a topic by comparing the observed co-occurrence of words within the topic to their expected co-occurrence in a reference corpus. C_V coherence measures the degree of coherence based on the exclusivity of the top words in a topic. Both UMass and V coherence metrics often indicate how well the topic model captures global patterns in the corpus (Lau & Baldwin, 2016).

C_{UCI} coherence and C_{NPMI} coherence, on the other hand, measure the degree of association between word pairs within a topic. C_{UCI} coherence calculates the pointwise mutual information (PMI) between the words in the topic, while C_{NPMI} normalizes PMI by dividing it by the negative logarithm of the probability of the word pair occurring together by chance. Both C_{UCI} and C_{NPMI} coherence metrics are based on the PMI, which often indicates how well the topic model captures local patterns in the corpus (Bouma, 2009a).

In conclusion, this experiment demonstrates that the HTM model outperforms the LDA model in the topic modeling of web content data and provides evidence of the benefits of the HTM model in learning coherent topics. These findings are useful for researchers in the field of web content analysis and topic modeling. Considering the structure of the web content indeed boosts the performance of the HTM model. These findings also answer the fourth research question of this study.

5.5. Experiment IV: Classification Framework

Cyber parental control aims to classify and filter web pages that contain objectionable topics. This experiment validates the achievement of the aim and evaluates the accuracy of the cyber parental control using several classifiers. The experiment uses HTM and LDA topic models to compare their classification accuracy when utilized in the framework. This experiment demonstrates the effectiveness of cyber parental control using the proposed HTM model in classifying objectionable web pages. The following sections describe the experiment, present the results, and discuss these results.

5.5.1. Experiment Aims and Description

This experiment evaluates cyber parental control using both HTM and LDA topic models. *Python* programming language with the help of Gensim (Rehurek, 2011), NLTK (Bird et al., 2009), scikit-learn (Pedregosa et al., 2011), pandas (McKinney, 2010), and spacy (Honnibal & Montani, 2017) libraries were used to implement the framework. Both topic models' implementation setup was similar to the previous experiment (as shown in Section 5.4). This experiment uses the default parameters' value of the scikit-learn library for the classifiers' implementation parameters.

The framework was validated and evaluated using *Dataset III* (as shown in Section 4.4.3). The classifiers adopted by this experiment are Support Vector Machine, Naïve Bayes, Random Forest, Logistic Regression, and K-Nearest Neighbor. This experiment

is evaluated by accuracy and F1 metrics. Using these metrics, the classification of each classifier using the HTM and LDA topic models given the number of topics $K \in \{1 \ldots, 100\}$ will then be compared in the following subsections. The following section discusses the results of both models and the indication of each result.

5.5.2. Experiment Result

The following subsections present each metric evaluation and comparison of the HTM and LDA topic models when utilized in the cyber parental control framework for classification using the objectionable ground truth dataset. Similar to the previous experiment, the results are discussed and visualized using a line chart. The chart's x-axis represents the number of topics, and the y-axis represents the accuracy metric score. The results of the framework using each model are plotted in a different line colour, which is explained in the legend of each graph. These results are then discussed in the following section (as shown in Section 5.5.3).

a) *Random Forest Classifier*. The analysis section compares the performance of HTM and LDA in a classification framework based on the RF classifier, as Table 5.3 tabulates. Across all topic numbers, the HTM model consistently outshines the LDA model, which sees its accuracy fluctuating between 75% and 78%. With 20 and 40 topics, the HTM model reaches its peak accuracy of over 94%, keeping an overall accuracy above 91.5% for all topic numbers. The F1 score trends mirror the accuracy results, with the HTM model scoring slightly higher and the LDA model moderately higher than their respective accuracy scores. The HTM model also hits its highest F1 score with 20 topics, at around 94%.

Table 5.3: Results of the classification framework based on the RF classifier using the LDA and HTM topic models

# of topics	LDA	HTM	# of topics	LDA	HTM
10	77.9%	92.5%	**10**	84.1%	93.6%

20	76.2%	**94.3%**
30	77.6%	92.5%
40	77.4%	**94.3%**
50	77.6%	93.4%
60	77.3%	92.5%
70	77.4%	92.8%
80	77.6%	91.4%
90	77.7%	91.1%
100	77.4%	91.1%

(a) Accuracy score results

20	82.9%	**95.0%**
30	83.8%	93.6%
40	83.8%	**95.0%**
50	83.9%	94.3%
60	83.8%	93.2%
70	83.9%	93.8%
80	84.0%	92.6%
90	84.0%	92.1%
100	83.8%	92.1%

(b) F1 score results

b) *K-Nearest Neighbor classifier*. The analysis section highlights the comparative performance of HTM and Latent LDA topic models in a classification framework using the KNN classifier, as Table 5.4 tabulates. The study suggests that the HTM topic model consistently outperforms the LDA model in both accuracy and F1 score across different topic numbers. Notably, HTM achieves its peak performance with 40 topics, securing an accuracy score close to 95% and an F1 score exceeding 95%. On the other hand, the LDA model's performance declines with an increase in topic numbers, peaking at 77% accuracy with 10 topics and dropping to around 70% with more than 40 topics. The F1 score of LDA was consistently less than 75% except when the topic number was 10.

Table 5.4: Evaluation results of the classification framework based on the KNN classifier using the LDA and HTM topic models

# of topics	LDA	HTM
10	77.7%	88.1%
20	72.3%	93.7%
30	70.3%	88.1%
40	70.6%	**94.6%**
50	70.3%	92.0%
60	69.6%	92.6%
70	69.8%	92.5%

# of topics	LDA	HTM
10	84.0%	90.3%
20	76.4%	94.5%
30	74.1%	90.3%
40	71.5%	**95.3%**
50	71.3%	93.1%
60	70.5%	93.5%
70	70.8%	93.5%

80	69.8%	93.4%
90	70.3%	93.4%
100	70.0%	94.0%

(a) Accuracy score results

80	71.0%	94.3%
90	71.5%	94.3%
100	70.9%	94.8%

(b) F1 score results

c) *Logistic Regression Classifier*. The analysis section offers a comparative assessment of the HTM and LDA topic models in a classification framework using the LR classifier, as Table 5.5 tabulates. It is found that the HTM topic model consistently exhibits superior performance over the LDA model in both accuracy and F1 score for the tested topic numbers. HTM's peak performance is achieved with 20 topics, securing approximately 89% accuracy and around 90% F1 score. Generally, the HTM model's accuracy remains above 84%, and the F1 score exceeds 86%. Conversely, the LDA model's accuracy fluctuates between 70% and 75% across the range of topic numbers, and its F1 score is slightly higher than its accuracy, yet it still lags behind HTM's results.

Table 5.5: Evaluation results of the classification framework based on the LR classifier using the LDA and HTM topic models

# of topics	LDA	HTM
10	73.0%	84.2%
20	72.4%	**89.0%**
30	71.2%	84.2%
40	71.6%	85.7%
50	71.1%	84.2%
60	72.2%	83.6%
70	72.8%	85.7%
80	70.9%	88.1%
90	74.1%	88.1%
100	72.0%	87.2%

(a) Accuracy score results

# of topics	LDA	HTM
10	77.5%	86.9%
20	76.3%	**90.0%**
30	76.0%	86.9%
40	74.4%	87.4%
50	73.9%	86.3%
60	76.9%	86.0%
70	77.4%	87.4%
80	73.8%	89.4%
90	78.3%	89.4%
100	76.7%	88.7%

(b) F1 score results

d) *Naïve Bayes Classifier*. The analysis section presents a comparison of the HTM and LDA topic models, as deployed in a classification framework using NB classifier, as Table 5.6 tabulates. According to the analysis, the HTM topic model consistently outperforms the LDA model in both accuracy and F1 score for the examined topic numbers. Specifically, the HTM model's accuracy sharply rises when the topic number exceeds 30, achieving its highest scores at 85% with 20 topics and 83.5% with 100 topics, and overall stays above 80% except for when the topic numbers are 10 and 30. In contrast, the LDA model's accuracy remains steady, fluctuating between 65% and 67.5% across different topic numbers, except for when the topic number is 20, which peaks at 72%. The F1 score results for the HTM model were higher than the accuracy results, with the highest F1 score observed at nearly 88% with 22 topics. However, like the accuracy results, the F1 scores for HTM significantly drop for 10 and 30 topics. LDA's F1 scores were consistently lower than its accuracy scores.

Table 5.6: Evaluation results of the classification framework based on the NB classifier using the LDA and HTM topic models

# of topics	LDA	HTM
10	65.7%	66.7%
20	72.2%	**84.8%**
30	65.8%	66.7%
40	66.5%	83.3%
50	66.3%	83.3%
60	65.5%	78.6%
70	65.7%	83.3%
80	66.1%	83.3%
90	65.5%	83.3%
100	65.0%	83.6%

(a) Accuracy score results

# of topics	LDA	HTM
10	63.9%	77.4%
20	76.2%	**86.7%**
30	63.6%	77.4%
40	64.2%	85.7%
50	63.6%	85.7%
60	63.1%	83.2%
70	63.4%	85.7%
80	63.8%	85.7%
90	62.4%	85.7%
100	62.1%	86.0%

(b) F1 score results

e) *Support Vector Machine Classifier*. This analysis section compares the performance of HTM and LDA topic models in a classification framework using the SVM

classifier, as Table 5.7 tabulates. The results show that the HTM topic model consistently scores higher than the LDA model in both accuracy and F1 scores across different topic numbers, except for when the topic numbers are 10 and 30. After the number of topics exceeds 30, the HTM model's accuracy score sharply rises, reaching its highest at nearly 87% with 90 topics. Excluding topics 10 and 30, HTM's accuracy stays above 83%. In contrast, the LDA model's accuracy fluctuates steadily between 75% and 77.5%. Both models' F1 scores surpass their accuracy results. Like its accuracy, the HTM model's highest F1 score is almost 90% with 90 topics, but it sees dramatic drops when the topic numbers are 10 and 30. Interestingly, the LDA model also achieves its highest F1 score, around 85%, with 90 topics.

Table 5.7: Evaluation results of the classification framework based on the SVM classifier using the LDA and HTM topic models

# of topics	LDA	HTM
10	76.6%	69.4%
20	77.0%	83.6%
30	75.5%	69.4%
40	76.5%	83.6%
50	76.2%	83.3%
60	76.5%	83.6%
70	76.6%	83.6%
80	76.6%	83.9%
90	78.1%	**86.9%**
100	76.5%	83.0%

(a) Accuracy score results

# of topics	LDA	HTM
10	83.4%	78.9%
20	83.5%	86.0%
30	82.3%	78.9%
40	83.3%	86.0%
50	83.1%	85.7%
60	83.2%	86.0%
70	83.3%	86.0%
80	83.4%	86.3%
90	84.3%	**88.5%**
100	83.2%	85.5%

(b) F1 score results

5.5.3. Experiment Discussion

The abovementioned results offer several indicators and findings. The overall performance of the cyber parental control framework using the HTM model is superior to its performance using the LDA model for both accuracy and F1 scores. The difference was substantial using all classifiers. The F1 scores usually resulted in higher accuracy

using the proposed HTM model. This indicates that the framework provides safer results using the proposed HTM model since it predicts fewer false negatives. The results also show that the framework's accuracy using the HTM model fluctuated when the topic number was less than 40, while it steadily increased when the topic number exceeded 40. In contrast, the framework accuracy using the LDA model slightly decreases when topics exceed 20-30. The high number of topics is beneficial to describe a webpage more accurately, as the framework's results using the proposed HTM model showed.

Scrutinizing the findings of the comparison, a few interesting observations are discovered. The two measurements and five classifiers show an improvement when using the HTM model. There was no drawback to using the proposed HTM compared to using LDA. The greatest improvement of the framework using the HTM model was about 30% of using the LDA based on the KNN classifier. The second improvement of using the HTM model was slightly more than 20% of using the LDA based on the RF classifier. The KNN and RF classifiers score the highest using the HTM model, with an accuracy of 93%. It is worth mentioning that among the used classifiers, the KNN and RF can handle high-dimensional data and a large number of features compared to others. Moreover, the improvement of the framework using the HTM model was more than 20% of using the LDA model based on all classifiers except the SVM. These results support the aforementioned conclusions that the HTM model learns better web content interpretable topics than the LDA model, and using it for classifying objectionable web content is more effective than the benchmark topic models.

5.6. Summary

This chapter applies the cyber parental control framework proposed in Chapter 4 and thoroughly evaluates its viability and strength. It begins by introducing three datasets. The dataset I contains conventional documents from Wikipedia and is used to evaluate the benchmark topic models. Dataset II contains web content-based data that evaluates

the proposed HTM topic model against benchmark topic models. Dataset III is a ground truth dataset containing about 2 million labelled web pages. This dataset is used to evaluate the cyber parental control framework. The evaluation measurements, including coherence and accuracy metrics used and the series of experiments, are then described.

Experiment I aims to ensure the validity of the ground truth labelling by using two different labelling sources and the kappa coefficient. The agreement score was almost perfect. Experiment II aims to demonstrate the limitations of the benchmark topic models; when applied to web data, the overall performance of these models dropped an average of 5 times and, in some cases, up to approximately 20 times lower than when applied to conventional data. Experiment III evaluates the effectiveness of the HTM model in discovering interpretable topics and term patterns in web content. The HTM model achieved an overall 36.5% improvement in topic coherence compared to the LDA. Finally, experiment IV validated the effectiveness of the proposed classification framework. The framework achieves an accuracy of 93% when using the proposed HTM topic model and a 30% improvement when using benchmark topic models.

CHAPTER 6: PROTOTYPE IMPLEMENTATION OF THE WEB CONTENT CLASSIFICATION FRAMEWORK

Chapter 6 presents the implementation and integration of the proposed cyber parental control framework into a functional prototype. It describes how the web-based prototype is developed, provides a comprehensive review of its primary features, and demonstrates its ease of use and simplicity. Finally, the advantages and limitations of the developed web-based prototype are discussed.

6.1. Requirements

This study presents a web application to prototype the proposed cyber parental control framework. The main functionality of the web application is to allow users to insert a specific URL to be classified. The following subsections detail the specifications of the web application prototype, including functional and non-functional requirements. Other requirements, such as user, business, and technical requirements, are yet to be specified in this prototyping.

6.1.1. Functional Requirements

The functional requirements of the web application prototype are as follows:

a) *Insert URL*. The user is able to insert any URL to be classified using cyber parental control, which is the main functionality of the prototype.

b) *User authentication*. This functionality is derived from the security perspective of the web application, and the user is able to register, log in, and log out to use the functionality mentioned above.

c) *False classification feedback submission*. In order to enhance the proposed cyber parental control, the prototype allows users to submit feedback when the classification of a specific URL results in a false classification.

6.1.2. Non-Functional Requirements

The prototype non-functional requirements are essential to guarantee that the web application meets its functional requirements and works properly based on these requirements. This study specifies the non-functional requirements of the prototype as follows:

a) *Performance*. This requirement ensures that the prototype responds quickly to the user's request for URL classification. This requirement includes the response time and throughput.
b) *Usability*. This requirement ensures that the prototype provides users with a clear and consistent user interface while hiding the complexity of the classification framework.
c) *Reliability*. This requirement ensures the prototype is always available and fully operational, with minimal downtime.
d) *Security*. In order to prevent misusing the web application, the prototype is only accessible to authorized users. Any user can register to the system and be authorized. This functionality protects the framework from security attacks and breaches.

6.2. Design and Architecture

This section details the design of the prototype and its architecture. It also illustrates the activity diagram and the sequence diagram of the main functionality of the web application. This section is a bridge between the requirements mentioned above and the implementation of the prototype. The following subsections briefly detail the high-level and low-level design of the prototype.

6.2.1. High-level design

The high-level design provides an overview of the prototype's architecture, major components, and interface. This overview helps provide a clear and complete

understanding of how the proposed cyber parental control prototype is designed to achieve its requirements.

The prototype is a web-based application that includes a few components and utilizes the RESTful API. These components are the listing component, scraping component, HTM topic model component, and classification component. Figure 6.1 illustrates the high-level architecture of the prototype and its components. Each of these components consists of an input, process, and output to fulfill its goal, as the following bullets describe:

a) *Listing component*. This component represents the first and second layers of the proposed cyber parental control framework. These lists are initially empty and will contain every URL that the other components have classified. This component is fast and lightweight, which helps to fulfill the performance requirement of the prototype.
b) *Scraping component*. When the URL is not found in the whitelist and blacklist, this component is considered the first step of the classification process. The component retrieves the source code of the webpage and extracts its contents based on predefined criteria. The output of this component is a JSON file that contains all the extracted features of the requested webpage.
c) *HTM topic model component*. This component is the content-based layer of the proposed cyber parental control. It first pre-processes the textual data of the input JSON file. The pre-processing of this component follows the steps defined earlier in this study (as shown in Section 4.1). It then runs the processed data into the HTM topic model. The model then generates topic vectors and exports them into a list of topics. Section 4.2.2 explained in detail the design of the used HTM topic model.
d) *Classification component*. This component inputs the topic vectors generated by the HTM topic model and uses a trained classifier model. The classifier has been chosen based on the performance comparison resulting from Section 5.5. The objectionable

ground truth dataset trains the model (Dataset III as shown in Section 4.4.3). The final output of this component is either 0 or 1 (0 as an unobjectionable webpage and 1 as an objectionable webpage).

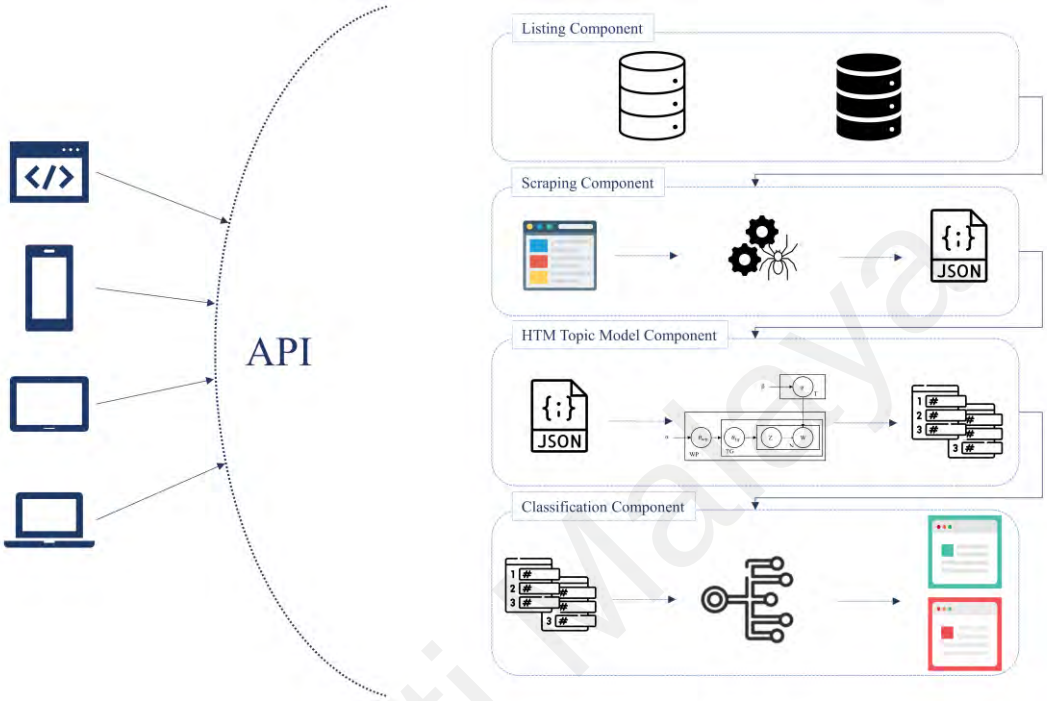

Figure 6.1: High-level component architecture

6.2.2. Low-level design

The low-level design provides a detailed description of how the prototype will be implemented using UML diagrams, algorithms, and pseudocodes. Since algorithms and pseudocodes have been addressed in the aforementioned sections (as shown in 4.1 and 4.2), this section focuses on the UML diagram for the main functional requirement of this prototype.

To further detail the interactions between components of the high-level design, Figure 6.2 illustrates the sequence diagram of classifying a requested URL. The sequence diagram addresses the order of these interactions and messages exchanges between the prototype's main components. Vertical lines, known as lifelines, stand in for the objects and components involved in the interactions, while horizontal arrows, known as messages, connect the objects' lifelines. The sequence begins when a client enters a target

URL and requests to classify it using the webpage interface. The returned result is either the class of the requested URL or an error response. The error might occur while accessing the URL webpage source code or the classification component.

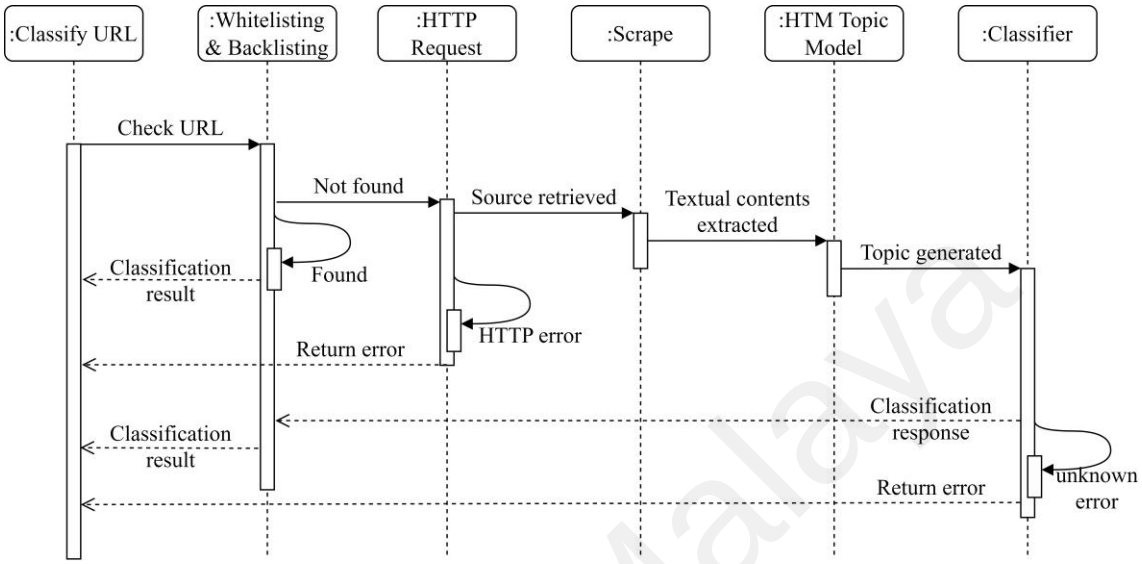

Figure 6.2: Sequence diagram of classifying a requested URL in the web topic model application

To further understand the prototype design of the major requirement, Figure 6.3 illustrates the activity diagram of requesting to classify a URL. An activity diagram represents the prototype's activities, the order in which activities are performed, and the decision points that control the flow. The activity begins at the front-end side by the client and might retrieve a different response to the request. This activity ensures that every request gets a response to achieve the reliability requirement of the prototype. There are four different responses that the user might receive. The successful response (represented by code 200) occurs when the prototype successfully classifies the requested URL. The second response occurs when the user is unauthorized to access the API of the cyber parental control (represented by code 401). The user can access it only after registering or logging in to the system. If a user sends a wrong request to the API, the response will be 400, representing a bad request error. The last response case is a retrieving data error (represented by the code 404), which occurs when the backend fails to retrieve or parse the source code of the requested URL webpage.

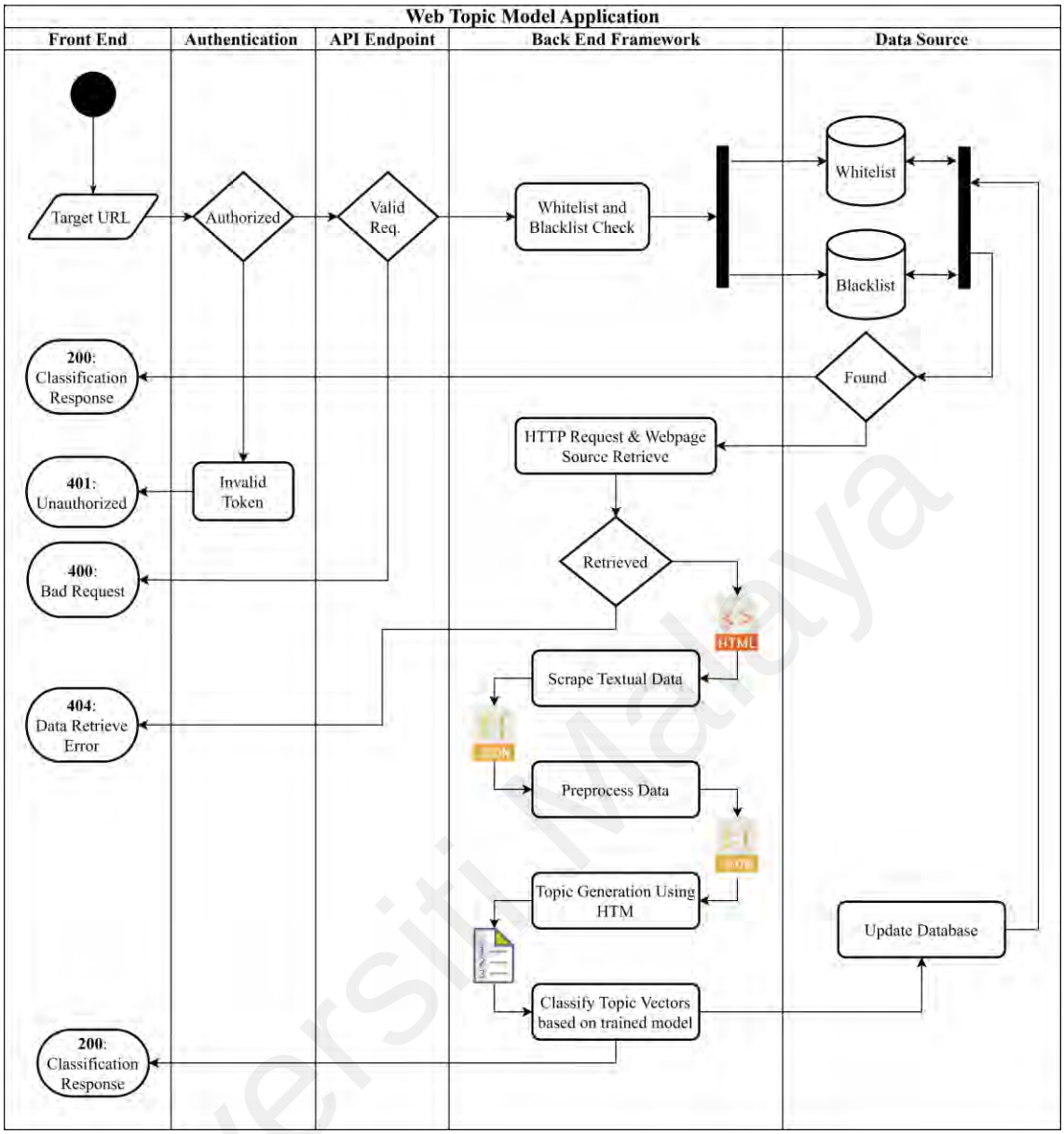

Figure 6.3: Activity diagram of classifying a requested URL in the web topic model application

6.3. Development and Implementation

This section translates the designed prototype into an implemented web-based application. The application contains two layers, the frontend layer and the backend layer. These layers communicate through an Application Programming Interface (API) RESTful architecture as follows:

a) *Front-end layer*. Also known as client-side, which refers to the development of the user interface and client-side logic of web applications. This prototype uses various web technologies such as HTML, CSS, and JavaScript to create the visual and

interactive elements of the web-based application that users interact with directly. The implementation also relies on a few libraries and frameworks, such as jQuery and Bootstrap, to ensure the efficiency of the web application.

b) *Back-end layer*. Refers to the server side of web application development, which manages the logic and data of the cyber parental control framework. The complexity of the backend is tremendous compared to the front end. The back end was developed using Python for its advantages over other programming languages. Various libraries were used to implement this layer: pandas, numpy, Beautifulsoup, Request, URLLib, scikit-learn, JSON, flask, flask_restful, flask_jwt_extended, waitress, spacy, gensim, and Pymongo.

c) *Database layer*. This prototype uses MongoDB, which is a popular NoSQL and open-source database. The document-oriented feature is the main advantage of this database, which stores data in semi-structured BSON (Binary JSON) format instead of traditional relational database tables and rows. The Pymongo library is an interface to connect to the Mongo Server.

6.4. Testing and Validation

This section tests and validates the implemented web-based application prototype's design and main functional requirements. Given that the proposed framework has been validated and evaluated thoroughly in Chapter 5, this section only includes frontend and API testing.

6.4.1. Frontend testing

Frontend testing refers to the practice of testing the user interface and user experience of the web-based application. The prototype UI consists of only one main page, which includes the main functionality. The main page allows the user to insert the target URL and submit it for classification. The validation of this functionality is as follows:

- The URL must not be empty.
- The URL must include the protocol (*http*:// or *https*://).
- The URL must include the network location.
- The URL must not include any spaces.

In case the user enters a URL that does not violate these validation criteria, the result of the main functionality will be one of the following:

a) *Objectionable*. This result appears when the framework classifies the web content of the target URL as objectionable. Figure 6.4 shows an example of where this result occurs.

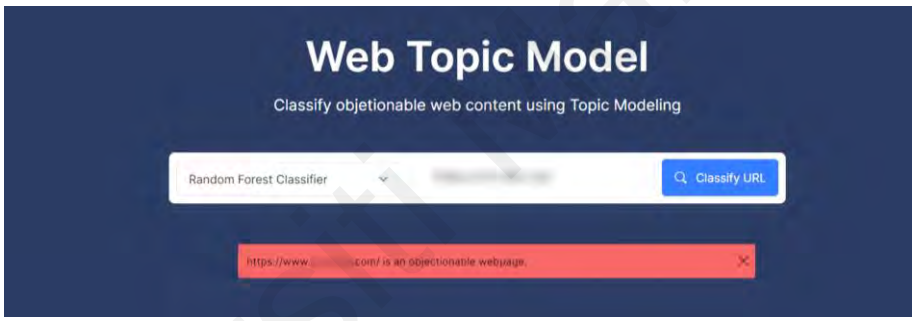

Figure 6.4: Objectionable classification result of the web-based application

b) *Unobjectionable*. This result appears when the framework classifies the web content of the target URL as unobjectionable. Figure 6.5 shows an example of where this result occurs.

Figure 6.5: Unobjectionable classification result of the web-based application

c) *Error*. This result appears if an error occurs, and a message will also appear to trace back to the source of the error. Figure 6.6 shows an example of where this result occurs.

Figure 6.6: Unknown error result of the web-based application

6.4.2. API testing

API testing focuses on testing individual endpoints of an API. The API of the prototype includes one endpoint of the main functionality. The structure of the endpoint is as follows:

$$GET\ /webtopicmodel/classify/?url = \{\}\&classifier = \{\}$$

The API responds with the data and the message generated by the backend framework. This study uses the Postman tool to test the API endpoint and response cases. The response status of this API endpoint can be as follows:

a) *200*. The request was successful, and the framework returned the target URL class. Figure 6.7 shows an example of a successful response and status of classifying a specific URL.

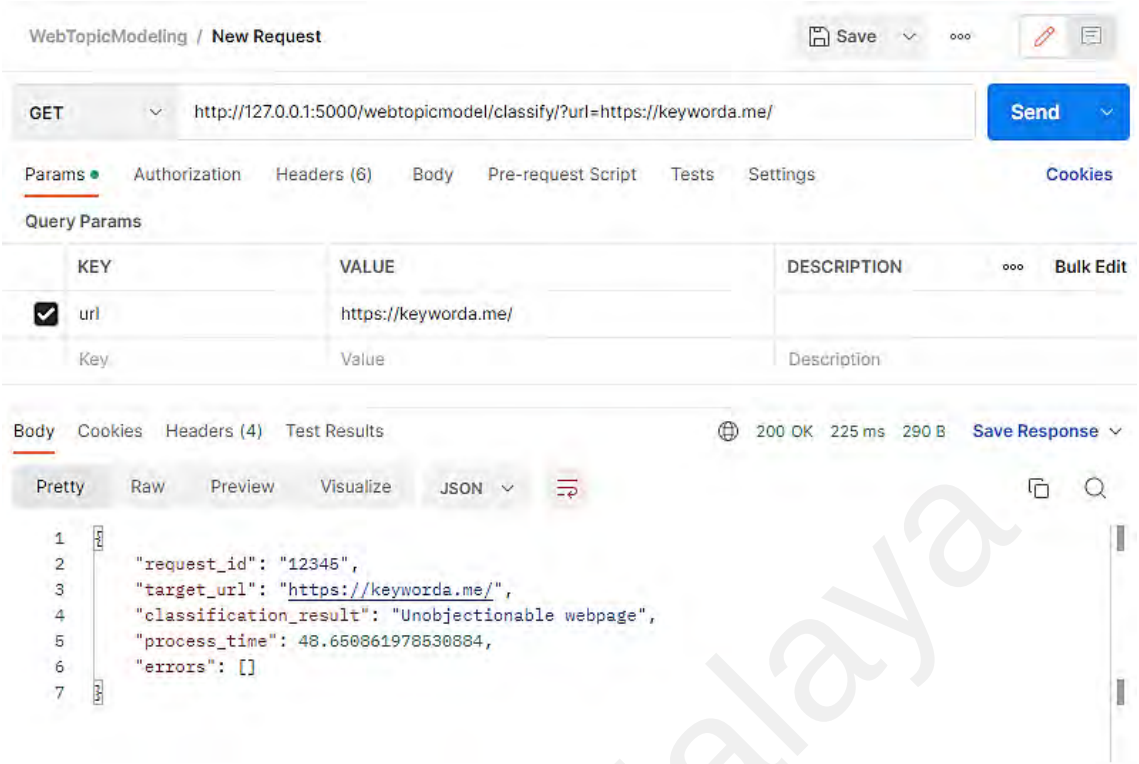

Figure 6.7: API response with status 200

b) *400*. The request was malformed or invalid due to typo errors or missing parameters. Figure 6.8 shows an example of an unsuccessful response due to missing the *URL* parameter from the request.

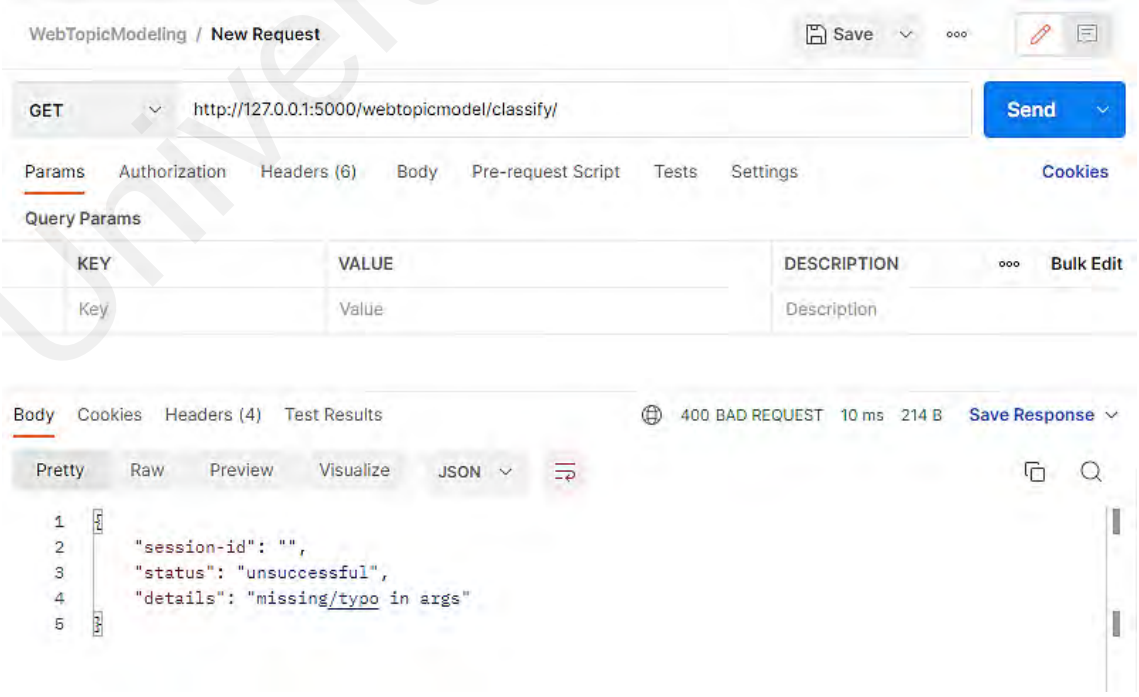

Figure 6.8: API response with status 400

c) *401*. The request requires authentication, and the client has not provided valid credentials. Figure 6.9 shows an example of an unsuccessful response due to an unauthorized request.

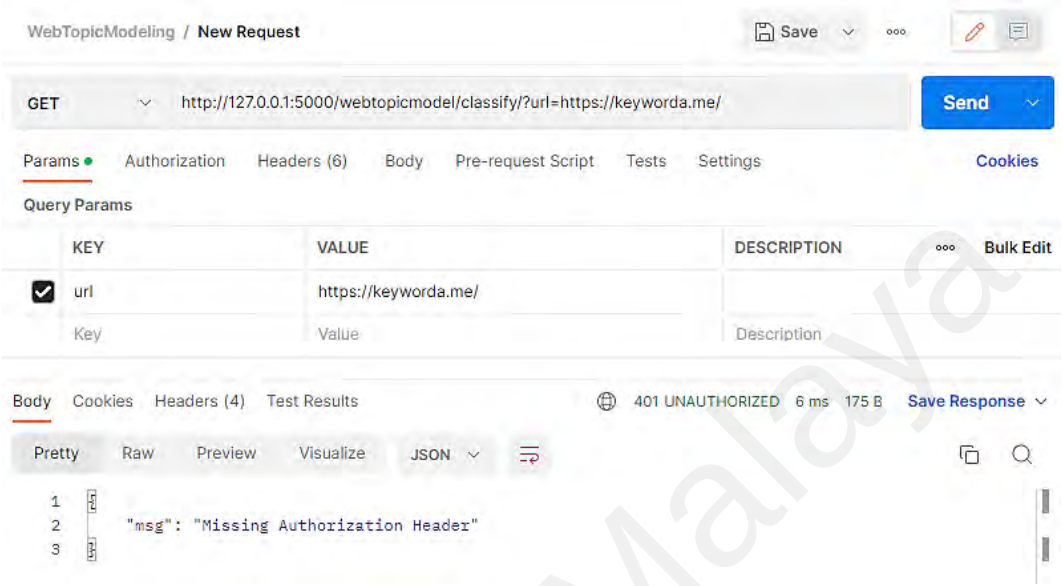

Figure 6.9: API response with status 401

d) *404*. The requested endpoint could not be found. Figure 6.10 shows an example of an unsuccessful response to an unknown endpoint request.

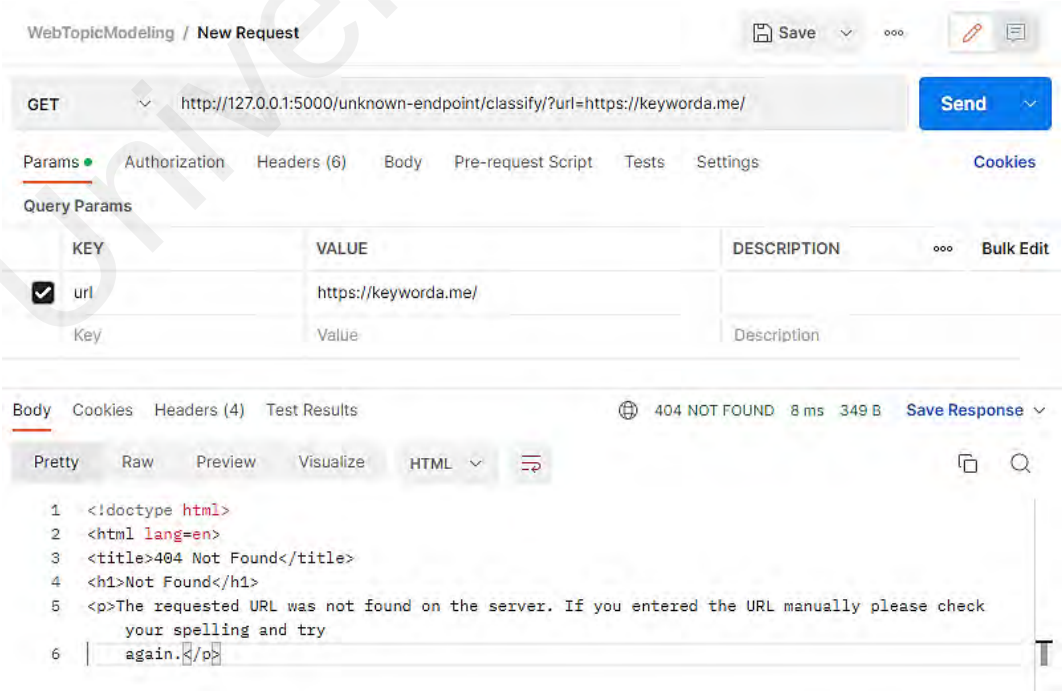

Figure 6.10: API response with status 404

e) *500*. The request has resulted in an unknown or unsupported error. Figure 6.11 shows an example of an unsuccessful response due to an internet connection error.

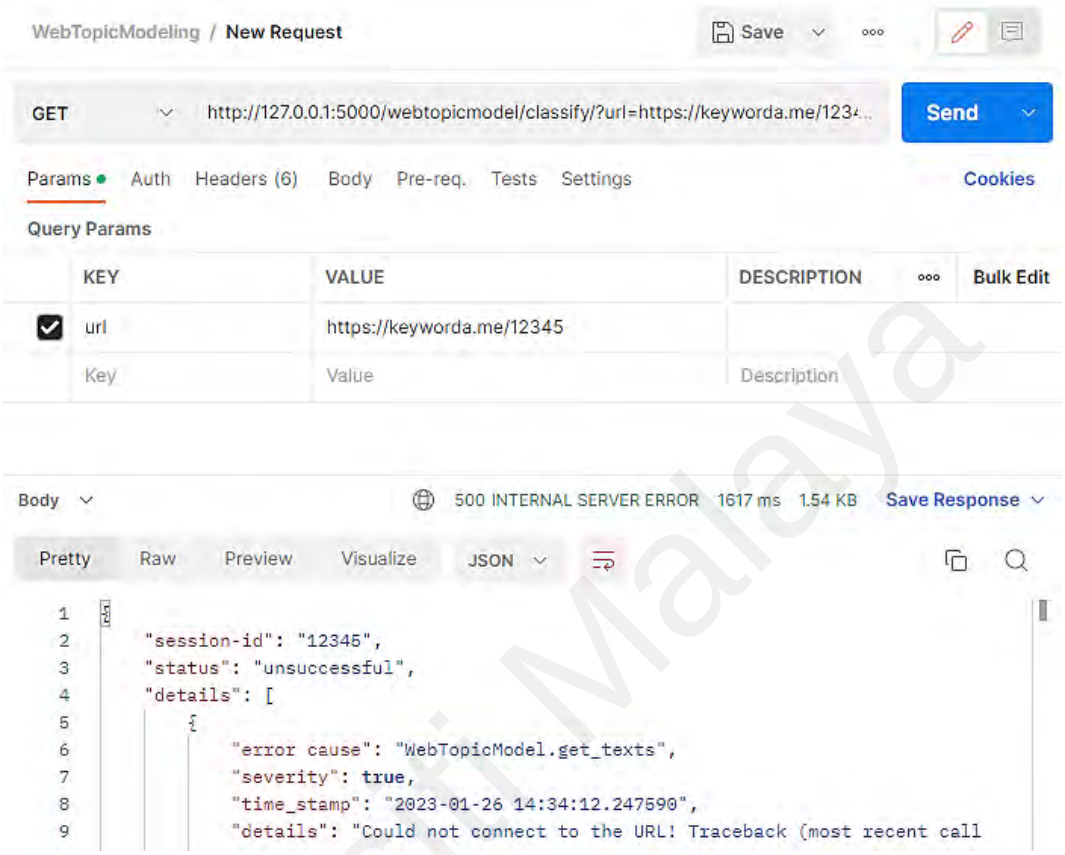

Figure 6.11: API response with status 500

6.5. Advantages and Limitations

The web-based prototype of cyber parental control applies the HTM topic model to web content data and classifies objectionable web pages. The implemented prototype offers the following advantages:

a) *Simplicity*. The prototype provides users with a single webpage interface to benefit from using the cyber parental control framework instead of using several web pages that complicate the application's usability. This advantage makes the prototype accessible to a broad spectrum of users, including non-technical individuals.

b) *Latency*. Effectiveness and fast response are essential for classification. The prototype validation showed its effectiveness and accuracy in classifying objectionable web pages based on the cyber parental control framework.

c) *Practical and economical*. The prototype is entirely built with open-source tools and frameworks, except for the compute engine, which Google Cloud hosts. It, therefore, provides a reasonable solution for enterprises to adopt web-based cyber parental control to achieve better web classification.

Although the prototype provides these significant advantages, it has some limitations. The lack of available resources poses the main limitation. The prototype's performance also depends on the server resources, such as the processing unit and memory. Low server specifications may hinder the overall performance of the prototype. In order to deliver a superior service in the future, it is essential to address this limitation by employing available alternative technologies and implementing consistent feature updates.

6.6. Summary

This chapter presents a prototype implementation of the proposed cyber parental control framework. It details the architecture and the key technologies used to construct the prototype. The prototype follows the distributed architecture, including a front end, a back end, and a database layer.

The principal features and functionalities of the prototype's components, including the Listing Layer, Web Scrapper, Pre-processor, Topic Modeling, and Classifier, are then discussed. These components abstract the framework's complexity to provide an easy-to-use application for the user. Finally, the chapter demonstrates the practical prototype and discusses its advantages and limitations.

This chapter translates all the modules proposed and evaluated in the previous chapters into a working prototype. The next chapter elaborates on the contributions of this study and the challenges faced and discusses future directions for research in this field.

CHAPTER 7: CONCLUSION

Chapter 7 concludes the study and its main findings and contributions. It also sheds light on the limitations faced in the study. The chapter provides prospects for future research on cyber parental control and web content classification. To this end, this study is a substantial effort to classify and filter objectionable web content using better means of topic modelling in order to provide children with a safe online environment free of harmful content.

7.1. Research Contributions

This study aimed to develop an effective cyber parental control method for accurately classifying and filtering objectionable web content using a novel web content topic modeling. In order to achieve this aim, the study began by investigating the cyber parental control field in general. The investigation addressed various paradigms of online safety approaches, web filtering methods, and web mining techniques. The study also reviewed the recent web content classification and filtering literature and the drawbacks in this domain field. Topic modeling is a sophisticated approach used for content classification and filtering. This study detailed the mathematical background of topic modeling and illustrated a taxonomy of probabilistic topic models. It also highlighted the software tools to implement and utilize topic models. Finally, the study benchmarked topic models based on their relativeness, robustness, and implementation.

This study proposed a cyber parental control framework utilizing the abovementioned background of the field. The proposed framework is an alternative solution to overcome the drawbacks of the existing solutions in the field of cyber parental control. The framework consists of three classification and filtering layers: whitelisting, blacklisting, and content-based filtering. The last layer is the main contribution of this study, which is

a topic model that learns interpretable topics from web content data effectively and accurately.

The study evaluated the proposed framework using several measurements based on three datasets. A series of experiments are conducted to ensure the validity of the ground truth dataset, the shortcomings of the benchmark models, the coherence of the proposed topic model, and the accuracy of the proposed framework. The proposed framework is implemented in a web application and deployed to classify objectionable web content.

The following summarizes this study's contributions throughout developing and deploying the proposed framework for cyber parental control.

a) *Conceptualization and definition of the terms "cyber parental control" and "objectionable web content"*. The study provided a clear definition of the term cyber parental control, which is a collection of parenting actions involving monitoring, controlling, and limiting children's activities on the web. The study also conceptualized objectionable web content terms as the textual or visual web content of a broad range of topics that certain internet users oppose, including, but not limited to, pornography, violence, drugs, hate, racism, homicide, gambling, and weapons.
b) *Review of web content classification studies*. The study provided a critical review of the latest studies in the domain. In Chapter 2, the study thoroughly reviewed content-based classification and filtering techniques. The review also critically analysed the web mining categories and techniques and addressed the strength and limitations of the recent studies in the field.
c) *An updated taxonomy of probabilistic topic models*. The literature review revealed the lack of taxonomy for the classification of the variety of probabilistic topic models in the literature as well as unclear benchmarking of those models, resulting in difficulties in evaluating the fitness for the purpose of the various models and comparing their

performance. This study contributed to solving the abovementioned issue by providing a methodical taxonomy of probabilistic topic models and benchmarking them for modeling topics in web content data.

d) *Demonstrate the drawbacks of existing topic models*. The study involved a systemic review of the performance of the existing topic models. The review demonstrated the shortcomings of the existing models when applied to web data compared to conventional document data (as shown in Section 5.3). This study aims to address the current research gap by presenting an improved topic model for web content-based data. This, together with the abovementioned achievements, fulfills the study's first objective.

e) *A reliable ground truth dataset of objectionable web data*. The study solved the unaddressed issues of the lack of a standard dataset in the current web content filtering studies and a ground truth dataset for objectionable and unobjectionable websites. A ground truth dataset containing about 2 million labelled web pages was created, validated, and made publicly available for developing and evaluating topic models for web content (Dataset III, as shown in Section 4.4.3).

f) *A coherent model for web topic classification*. Provided the insufficiency of benchmark topic models on web content data for web content modeling, this study proposed an innovative topic model (called HTM) to learn interpretable topics in web content data (as shown in Section 4.1). The proposed HTM topic model took into consideration the HTML tags to understand the structure of web pages and resulted in a substantial enhancement in topic coherence, as illustrated in Experiment III (as shown in Section 5.4). This fulfills the second objective of this study.

g) *An accurate framework for cyber parental control*. Based on the proposed solution for modeling topics on web content data (the HTM topic model), this study developed an effective and accurate framework for classifying and filtering objectionable web

content (as shown in Section 4.2). The framework employed a multistep approach, including the URL whitelist and blacklist methods as the first and second filter layers and the proposed HTM topic model as the third layer. The performance of the proposed framework was evaluated using five different classifiers. This contribution fulfills the third objective of this study. This contribution, together with contribution (f), also satisfies the fourth objective of this study.

h) *Implementation of the proposed framework*. This study designed and developed a web application as a proof-of-concept for the proposed cyber parental control framework. The application aims to demonstrate the framework's effectiveness in classifying objectionable web pages. This study detailed the development procedure, including designing and implementing the frontend, backend, and database modules. The web-based system provides practical and easy-to-use access for users by abstracting the technical complexity of the framework architecture.

The abovementioned achievements provide convincing evidence that this study has fulfilled its main aim and objectives outlined in Chapter 1.

7.2. Limitations of the Study

Despite the substantial achievements detailed in the previous sections, a few limitations were encountered during this study. These limitations represent potential opportunities for future research and are discussed below.

a) *Languages*. This study focused on the English language as the source of the input data for the classification framework, as it represents 57.1% of web content. The HTM topic model and the framework have not been tested for other languages. However, the conceptual model is hypothetically applicable to other languages, and therefore, future works could evaluate the HTM topic model based on languages like Arabic, French, German, and others.

b) *Multi-class classification*. The CPC framework predominantly addresses binary classification, which discerns between objectionable and non-objectionable web content. This binary approach might overlook the importance of various levels of objectionable content or multiple thematic categorizations. It is recommended for future studies to delve into the intricacies of multi-class classification, where web content can be categorized into multiple predefined classes, offering a more comprehensive and customized content filtering mechanism.

c) *Visual content*. This study focuses only on textual web content and leaves out of its scope visual web content. The proposed framework for cyber parental control may not be able to filter objectionable web pages if they contain only visuals without textual content.

d) *Limitation of static analysis*. During the experiment, the data pre-processing step stopped responding due to RAM limitations. This issue occurred on datasets with a very large number of web pages and textual tags. Nevertheless, low-size datasets ran smoothly during the data pre-processing step of the framework.

e) *Processing time evaluation*. The HTM topic model was evaluated based on topic coherence metrics, while the cyber parental control framework was based on accuracy and F1 metrics. The evaluation metrics exclude processing time, which can be addressed in future studies.

f) *Usability of the prototype*. This study demonstrated the effectiveness of the web-based prototype in classifying objectionable web pages. The conceptual design of the prototype was addressed through several snapshots that adequately illustrate the web application in action. Nonetheless, the usability of the web application and its modules is outside the study scope and can be thoroughly tested in future research in all aspects (reliability, sustainability, and response time).

g) *Dynamic web content*. The study mainly addressed static content for whitelist and blacklist creation. The challenges posed by dynamic, frequently updated content, such as live feeds or real-time updates, were not extensively covered. Future research should focus on the real-time classification of such dynamic content for a more comprehensive filtering system.

7.3. Future Work and Directions

There are several directions for future research building upon the achievement of the current study to address some limitations and enhance its applicability and adaptability. Those can be summarized as follows.

a) *Include a variety of languages*. Other languages, such as Arabic, Spanish, Chinese, and French, are important to account for in web content classification and filtering. Future works could include these languages to build an enhanced Multi-Lingual HTM topic model (ML-HTM). This model would have various applications in different regions. The challenge of such work is finding corpora and datasets for each language.

b) *Visual content classification*. The second important type of web content is visuals, including images and videos. Accounting for visual and textual content will result in more coherent topics and accurate classification. Future works could incorporate topic models for visual content along with the HTM textual topic model to increase the efficiency and effectiveness of cyber parental control frameworks.

c) *Apply the model to various topics*. Given its scope, this study considered only the objectionable versus unobjectionable categorization of topics. However, as with any other topic model, the HTM topic model also can work on other topics. This characteristic opens opportunities for future works to apply the HTM topic model to different topic categories, such as sports, politics, academics, business, health, and many more.

d) *Utilize web structure mining*. Web structure mining identifies graph patterns of websites by utilising hyperlinks. The structure helps identify parents, siblings, and children's web pages on a specific webpage. This knowledge is worth utilizing to get a comprehensive picture of the webpage and can result in more accurate classification. Although this work might increase the complexity of the model, it could help discover new future directions and prospects for research in the field of web topic modeling.

e) *Address social media*. There is no doubt that social media are the trend now, making their content more important to address. This study shows that the existing topic models perform relatively well when applied to conventional documents but underperform on web content data. This study proposed a topic model that overcomes this shortcoming and performs well on web content, breaking the chain of applying topic models only on conventional data sources such as documents and articles and opening the door for developing topic models for practical and trending applications. Examples of such applications include social media, Non-Fungible Tokens (NFT), and future trends that might contain unconventional data sources. These applications and data sources are worthy of investigation in future work in order to propose comprehensive topic models.

f) *Building commercial/non-profit projects*. This study has designed, developed, and prototyped the proposed framework using the proposed HTM topic model. In future work, the developed prototype can be enhanced and deployed in sophisticated commercial or non-profit projects for learning and generating topics of web pages. Ideas for such projects include finding similar websites, analyzing websites for SEO, web indexing and ranking, and filtering systems.

g) *Expand the benchmark*. The benchmark of this study is limited to topic models that fit into the study's selection criteria. However, it might be useful to include the

comparison with non-probabilistic topic modeling approaches such as the BERTopic model and recent word embedding models.

7.4. Summary

The Internet makes a massive amount of web content, including objectionable materials, available for users. These objectionable contents, such as pornography, drugs, weapons, gambling, violence, and hatred, can pose severe problems for Internet users, especially children. This study has presented an effective and accurate framework for classifying these objectionable web contents that utilizes a coherent topic model. The study has demonstrated that using existing topic models for this task was insufficient due to the fact the conventional models neglect the unique structure of web content. The study has also created a ground truth dataset to help future studies make a fair and coherent comparison of the various frameworks proposed in the field of cyber parental control. The contributions of this study have been achieved by designing, developing, evaluating, and prototyping the cyber parental control framework using the HTM topic model.

REFERENCES

Ahmadi, A., Fotouhi, M., & Khaleghi, M. (2011). Intelligent classification of web pages using contextual and visual features. *Applied Soft Computing*, *11*(2), 1638-1647. https://doi.org/10.1016/j.asoc.2010.05.003

Ahmed, W., & Jameel, N. G. M. (2022). Malicious URL Detection Using Decision Tree-based Lexical Features Selection and Multilayer Perceptron Model. *UHD Journal of Science and Technology*, *6*(2), 105-116.

AlAgha, I. (2022). Leveraging Knowledge-Based Features with Multilevel Attention Mechanisms for Short Arabic Text Classification. *IEEE Access*.

Aletras, N., & Stevenson, M. (2013). Evaluating topic coherence using distributional semantics. Proceedings of the 10th International Conference on Computational Semantics (IWCS 2013)–Long Papers,

Alghamdi, H., & Selamat, A. (2015). Topic modelling used to improve Arabic web pages clustering. 2015 International Conference on Cloud Computing (ICCC),

Ali, F., Khan, P., Riaz, K., Kwak, D., Abuhmed, T., Park, D., & Kwak, K. S. (2017). A fuzzy ontology and SVM–based Web content classification system. *IEEE Access*, *5*, 25781-25797.

Alkhodair, S. A., Fung, B. C., Rahman, O., & Hung, P. C. (2018). Improving Interpretations of Topic Modeling in Microblogs. *Journal of the Association for Information Science and Technology*, *69*, 528-540. https://doi.org/10.1002/asi

Almatrooshi, F., Alhammadi, S., Salloum, S. A., & Shaalan, K. (2022). Text and web content mining: a systematic review. Proceedings of International Conference on Emerging Technologies and Intelligent Systems: ICETIS 2021 (Volume 1),

AlSumait, L., Barbará, D., & Domeniconi, C. (2008). On-line lda: Adaptive topic models for mining text streams with applications to topic detection and tracking. 2008 eighth IEEE international conference on data mining,

Altarturi, H. (2022). *CrawlerScraper*. https://github.com/hamzatartori/CrawlerScraper

Altarturi, H., Saadoon, M., & Anuar, N. B. (2023). Web content topic modeling using LDA and HTML tags. *PeerJ Computer Science*. https://doi.org/DOI 10.7717/peerj-cs.1459

Altarturi, H. H. M., Saadoon, M., & Anuar, N. B. (2020). Cyber parental control: A bibliometric study. *Children and Youth Services Review*, *116*. https://doi.org/10.1016/j.childyouth.2020.105134

Altay, B., Dokeroglu, T., & Cosar, A. (2019). Context-sensitive and keyword density-based supervised machine learning techniques for malicious webpage detection. *Soft Computing*, *23*(12), 4177-4191.

Anami, B. S., Wadawadagi, R. S., & Pagi, V. B. (2014). Machine Learning Techniques in Web Content Mining: A Comparative Analysis. *Journal of Information & Knowledge Management*, *13*(01). https://doi.org/10.1142/s0219649214500051

Andriansyah, M., Purwanto, I., Subali, M., Sukowati, A. I., Samos, M., & Akbar, A. (2017, November). *Developing Indonesian corpus of pornography using simple NLP-text mining (NTM) approach to support government anti-pornography program* Informatics and Computing (ICIC),

Arora, S., Ge, R., Halpern, Y., Mimno, D., Moitra, A., Sontag, D., Wu, Y., & Zhu, M. (2013). A practical algorithm for topic modeling with provable guarantees. International Conference on Machine Learning,

Artene, C.-G., Vecliuc, D.-D., Tibeică, M. N., & Leon, F. (2022). An Experimental Study of Convolutional Neural Networks for Functional and Subject Classification of Web Pages. *Vietnam Journal of Computer Science*, 1-19.

Asdaghi, F., & Soleimani, A. (2019). An effective feature selection method for web spam detection. *Knowledge-Based Systems*, *166*, 198-206.
Asdaghi, F., Soleimani, A., & Zahedi, M. (2020). A novel set of contextual features for web spam detection. *International Journal of Nonlinear Analysis and Applications*, *11*(1), 321-339.
Aunola, K., Ruusunen, A.-K., Viljaranta, J., & Nurmi, J.-E. (2015). Parental affection and psychological control as mediators between parents' depressive symptoms and child distress. *Journal of Family Issues*, *36*(8), 1022-1042.
bab2min, D. F., & Jonathan Schneider. (2021). bab2min/tomotopy: 0.12.1. *Zenodo*. https://doi.org/https://doi.org/10.5281/zenodo.5000206
BeautifulSoup Library. https://pypi.org/project/beautifulsoup4 , urldate = 2022-04-28
Berardi, G., Esuli, A., Fagni, T., & Sebastiani, F. (2015). *Classifying websites by industry sector* Proceedings of the 30th Annual ACM Symposium on Applied Computing - SAC '15,
Berkson, J. (1944). Application of the logistic function to bio-assay. *Journal of the american statistical association*, *39*(227), 357-365.
Bijalwan, V., Kumari, P., Pascual, J., & Semwal, V. B. (2014). Machine learning approach for text and document mining. *arXiv preprint arXiv:1406.1580*.
Bird, S., Klein, E., & Loper, E. (2009). Natural Language Toolkit (NLTK) (Version 3.8.1) [Software]. *Natural language processing with Python: analyzing text with the natural language toolkit*. O'Reilly Media, Inc.
Blei, D., & Lafferty, J. (2006). Correlated topic models. *Advances in neural information processing systems*, *18*, 147.
Blei, D. M., & Lafferty, J. D. (2006). Dynamic topic models. Proceedings of the 23rd international conference on Machine learning,
Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, *3*, 993-1022.
Bouma, G. (2009a). Normalized (pointwise) mutual information in collocation extraction. *Proceedings of GSCL*, *30*, 31-40.
Bouma, G. (2009b). Normalized (pointwise) mutual information in collocation extraction. *Proceedings of GSCL*, 31-40.
Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5-32.
Capital, V. (2018). *VP Capital and Larnabel Ventures Announce $2 Million Investment in Smart Parental Control Startup FaceMetrics*. prnewswire. Retrieved 2019 from https://www.prnewswire.com/news-releases/vp-capital-and-larnabel-ventures-announce-2-million-investment-in-smart-parental-control-startup-685539451.html
Caulkins, J. P., Ding, W., Duncan, G., Krishnan, R., & Nyberg, E. (2006). A method for managing access to web pages: Filtering by Statistical Classification (FSC) applied to text. *Decision Support Systems*, *42*(1), 144-161. https://doi.org/10.1016/j.dss.2004.11.015
Chapman, D. B. (1992). Network (In) Security Through IP Packet Filtering. *USENIX Summer*.
Chau, M., & Chen, H. (2008). A machine learning approach to web page filtering using content and structure analysis. *Decision Support Systems*, *44*(2), 482-494. https://doi.org/10.1016/j.dss.2007.06.002
Chen, F., & Zhou, Y. H. (2014). A Tag-based Improved LDA and Web Page Clustering Analysis. Applied Mechanics and Materials,
Chen, T. H., Thomas, S. W., Nagappan, M., & Hassan, A. E. (2012, June). *Explaining software defects using topic models* Mining Software Repositories (MSR),
Chen, Z., & Liu, B. (2014). Topic modeling using topics from many domains, lifelong learning and big data. International conference on machine learning,

Chiang, I.-J., Liu, C. C.-H., Tsai, Y.-H., & Kumar, A. (2015). Discovering latent semantics in web documents using fuzzy clustering. *IEEE Transactions on Fuzzy Systems*, *23*(6), 2122-2134.

Choi, J. (2019). Tomotopy (Version 0.12.5) [Library]. *Available at https://github.com/bab2min/tomotopy*.

Chung, K., Yoo, H., Choe, D., & Jung, H. (2019). Blockchain network based topic mining process for cognitive manufacturing. *Wireless Personal Communications*, *105*(2), 583-597.

Cohn, D., & Hofmann, T. (2000). The missing link-a probabilistic model of document content and hypertext connectivity. *Advances in neural information processing systems*, *13*.

Conneau, A., & Lample, G. (2021). MUSE (Version 1.0.0) [Library]. *Available at https://pypi.org/project/pythonMuse/*.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273-297.

Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE transactions on information theory*, *13*(1), 21-27.

Cramer, J. S. (2002). The origins of logistic regression.

Dai, Y., Wang, S., Chen, X., Xu, C., & Guo, W. (2020). Generative adversarial networks based on Wasserstein distance for knowledge graph embeddings. *Knowledge-Based Systems*, *190*, 105165.

Deibert, R., Palfrey, J., Rohozinski, R., & Zittrain, J. (2010). *Access controlled: The shaping of power, rights, and rule in cyberspace*. the MIT Press.

Demirkıran, F., Çayır, A., Ünal, U., & Dağ, H. (2020). Website category classification using fine-tuned BERT language model. 2020 5th International Conference on Computer Science and Engineering (UBMK),

Dilip Patel, A., & Pandya, V. N. (2017, 2017). Web page classification based on context to the content extraction of articles. 2017 2nd International Conference for Convergence in Technology (I2CT),

Ding, D., Han, Q.-L., Wang, Z., & Ge, X. (2019). A survey on model-based distributed control and filtering for industrial cyber-physical systems. *Ieee Transactions on Industrial Informatics*, *15*(5), 2483-2499.

Dishion, T. J., & McMahon, R. J. (1998). Parental monitoring and the prevention of problem behavior: A conceptual and empirical reformulation. In *Drug abuse prevention through family interventions* (Vol. 177, pp. 229).

Doan, T.-N., & Hoang, T.-A. (2021). Benchmarking neural topic models: An empirical study. Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021,

Duan, J., & Zeng, J. (2013). Web objectionable text content detection using topic modeling technique. *Expert Systems with Applications*, *40*(15), 6094-6104. https://doi.org/10.1016/j.eswa.2013.05.032

Duan, J., Zeng, J., & Zhang, S. (2012, August). *Hierarchical semantic model for objectionable Web text content detection* Anti-Counterfeiting, Security and Identification (ASID),

Dwivedi, S. K., & Rawat, B. (2015). A review paper on data preprocessing: a critical phase in web usage mining process. 2015 International Conference on Green Computing and Internet of Things (ICGCIoT),

Ehikioya, S. A., & Zeng, J. (2021). Mining web content usage patterns of electronic commerce transactions for enhanced customer services. *Engineering Reports*, *3*(11), e12411.

Eickhoff, C., Polajnar, T., Gyllstrom, K., Torres, S. D., & Glassey, R. (2011). Web Search Query Assistance Functionality for Young Audiences. In P. Clough, C. Foley, C.

Gurrin, G. J. F. Jones, W. Kraaij, H. Lee, & V. Mudoch, *Advances in Information Retrieval* Berlin, Heidelberg.
Elsaesser, C., Russell, B., Ohannessian, C. M., & Patton, D. (2017). Parenting in a digital age: A review of parents' role in preventing adolescent cyberbullying. *Aggression and Violent Behavior*, *35*, 62-72. https://doi.org/10.1016/j.avb.2017.06.004
Etzioni, O. (1996). The World-Wide Web: quagmire or gold mine? *Communications of the ACM*, *39*(11), 65-68. https://doi.org/10.1145/240455.240473
Feroz, M. N., & Mengel, S. (2015). *Phishing URL Detection Using URL Ranking* 2015 IEEE International Congress on Big Data,
Figueiredo, F., Pinto, H., Belém, F., Almeida, J., Gonçalves, M., Fernandes, D., & Moura, E. (2013). Assessing the quality of textual features in social media. *Information Processing & Management*, *49*(1), 222-247. https://doi.org/10.1016/j.ipm.2012.03.003
Fu, Q., Zhuang, Y., Gu, J., Zhu, Y., & Guo, X. (2021). Agreeing to Disagree: Choosing Among Eight Topic-Modeling Methods. *Big Data Research*, *23*. https://doi.org/10.1016/j.bdr.2020.100173
Fuertes, W., Quimbiulco, K., Galárraga, F., & García-Dorado, J. L. (2015). On the development of advanced parental control tools. 2015 1st International Conference on Software Security and Assurance (ICSSA),
Fujimura, K., Inoue, T., & Sugisaki, M. (2005). The eigenrumor algorithm for ranking blogs. WWW 2005 2nd Annual Workshop on the Weblogging Ecosystem,
GeoIP Database. https://geolocation-db.com , urldate = 2022-04-28
Google. (2019). *Chrome & your child's Google Account*. Retrieved 26/12/2019 from
Griffiths, T. L., Jordan, M. I., Tenenbaum, J. B., & Blei, D. M. (2004). Hierarchical topic models and the nested chinese restaurant process. *Advances in neural information processing systems*, 17-24.
Griffiths, T. L., & Steyvers, M. (2004). Finding scientific topics. *Proceedings of the National academy of Sciences*, *101*(suppl_1), 5228-5235.
Guo, F., Metallinou, A., Khatri, C., Raju, A., Venkatesh, A., & Ram, A. (2018). Topic-based evaluation for conversational bots. *arXiv preprint arXiv:1801.03622*.
Gupta, P., Chaudhary, Y., Buettner, F., & Schütze, H. (2019). Document informed neural autoregressive topic models with distributional prior. Proceedings of the AAAI Conference on Artificial Intelligence,
Hai, Z., Cong, G., Chang, K., Cheng, P., & Miao, C. (2017). Analyzing sentiments in one go: A supervised joint topic modeling approach. *IEEE Transactions on Knowledge and Data Engineering*, *29*(6), 1172-1185.
Hajjem, M., & Latiri, C. (2017). Combining IR and LDA Topic Modeling for Filtering Microblogs. *Procedia Computer Science*, *112*, 761-770.
Hammami, M., Chahir, Y., & Chen, L. (2006). WebGuard: a Web filtering engine combining textual, structural, and visual content-based analysis. *IEEE Transactions on Knowledge and Data Engineering*, *18*(2), 272-284. https://doi.org/10.1109/tkde.2006.34
Hariri, R. H., Fredericks, E. M., & Bowers, K. M. (2019). Uncertainty in big data analytics: survey, opportunities, and challenges. *Journal of Big Data*, *6*(1), 1-16.
Heinrich, G. (2005). *Parameter estimation for text analysis*.
Hilal, S., & Gupta, N. (2013). Role of Web Content Mining in Kids' based Mobile Search. *International Journal of Computer Applications*, *62*(6), 12-17. https://doi.org/10.5120/10083-4707
Hofmann, T. (1999). Probabilistic latent semantic indexing. Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval,

Honnibal, M., & Montani, I. (2017). spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing.
Hussain, M., Ahmed, M., Khattak, H. A., Imran, M., Khan, A., Din, S., Ahmad, A., Jeon, G., & Reddy, A. G. (2018). Towards ontology-based multilingual URL filtering: a big data problem. *The Journal of Supercomputing*, *74*(10), 5003-5021. https://doi.org/10.1007/s11227-018-2338-1
Ibrahim, M. E., Yang, Y., Ndzi, D. L., Yang, G., & Al-Maliki, M. (2018). Ontology-based personalized course recommendation framework. *IEEE Access*, *7*, 5180-5199.
Ibrahim, S. (2016a). Causes of socioeconomic cybercrime in Nigeria. IEEE International Conference on Cybercrime and Computer Forensic (ICCCF),
Ibrahim, S. (2016b). Social and contextual taxonomy of cybercrime: Socioeconomic theory of Nigerian cybercriminals. *International Journal of Law, Crime and Justice*, *47*, 44-57. https://doi.org/10.1016/j.ijlcj.2016.07.002
Ihekoronye, V. U., Ajakwe, S. O., Kim, D.-S., & Lee, J. M. (2022). Cyber Edge Intelligent Intrusion Detection Framework For UAV Network Based on Random Forest Algorithm. 2022 13th International Conference on Information and Communication Technology Convergence (ICTC),
Jacob, V. S., Krishnan, R., & Ryu, Y. U. (2007). Internet content filtering using isotonic separation on content category ratings. *ACM Transactions on Internet Technology*, *7*(1), 1-es. https://doi.org/10.1145/1189740.1189741
Jain, S., Cohen, A. K., Paglisotti, T., Subramanyam, M. A., Chopel, A., & Miller, E. (2018). School climate and physical adolescent relationship abuse: Differences by sex, socioeconomic status, and bullying. *J Adolesc*, *66*, 71-82. https://doi.org/10.1016/j.adolescence.2018.05.001
Jicheng, W., Yuan, H., Gangshan, W., & Fuyan, Z. (1999). Web Mining: Knowledge Discovery on the Web. In Systems, Man, and Cybernetics,
Jo, Y., & Oh, A. H. (2011). Aspect and sentiment unification model for online review analysis. Proceedings of the fourth ACM international conference on Web search and data mining,
Kaptur, V., & Kniaziev, O. (2019). Method of adaptive complex Internet content filtering. 2019 International Conference on Information and Telecommunication Technologies and Radio Electronics (UkrMiCo),
Kelly, A., & Johnson, M. A. (2021). Investigating the statistical assumptions of Naïve Bayes classifiers. 2021 55th annual conference on information sciences and systems (CISS),
Khan, N. A., Khan, A., Ahmad, M., Shah, M. A., & Jeon, G. (2021). URL filtering using big data analytics in 5G networks. *Computers and Electrical Engineering*, *95*, 107379.
Kiddle. (2019). *How is Kiddle designed specifically for kids?* Retrieved 5 March from https://www.kiddle.co/about.php
kidrex. (2019). *What is KidRex and how does it work?* kidrex. Retrieved 5 March from https://www.alarms.org/kidrex/parents/about.html
Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, *46*, 604-632.
Kosala, R., & Blockeel, H. (2000). Web mining research. *2*(1), 1-15. https://doi.org/10.1145/360402.360406
Kosala, R., & Blockeel, H. (2000). Web mining research: A survey. *ACM Sigkdd Explorations Newsletter*, *2*(1), 1-15.
Kotenko, I., Chechulin, A., Shorov, A., & Komashinsky, D. (2014). Analysis and evaluation of web pages classification techniques for inappropriate content blocking. Industrial Conference on Data Mining,

Krishnani, D., Kumari, A., Dewangan, A., Singh, A., & Naik, N. S. (2019). Prediction of coronary heart disease using supervised machine learning algorithms. TENCON 2019-2019 IEEE Region 10 Conference (TENCON),

Kumar, I., Dogra, K., Utreja, C., & Yadav, P. (2018). A comparative study of supervised machine learning algorithms for stock market trend prediction. 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT),

Kumari, M., & Soni, S. (2017). A review of classification in web usage mining using K-nearest neighbour. *Advances in Computational Sciences and Technology*, *10*(5), 1405-1416.

Kumbhar, R. (2012). Classification and its uses. In *Library Classification Trends in the 21st Century* (pp. 7-24). https://doi.org/10.1016/b978-1-84334-660-9.50002-3

Lafferty, J., & Blei, D. (2005). Correlated topic models. *Advances in neural information processing systems*, *18*.

Lau, J. H., & Baldwin, T. (2016). An empirical evaluation of doc2vec with practical insights into document embedding generation. *arXiv preprint arXiv:1607.05368*.

Lau, J. H., Newman, D., & Baldwin, T. (2014). Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality. Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics,

Law, D. M., Shapka, J. D., & Olson, B. F. (2010). To control or not to control? Parenting behaviours and adolescent online aggression. *Computers in Human Behavior*, *26*(6), 1651-1656. https://doi.org/10.1016/j.chb.2010.06.013

Lee, L.-H., Juan, Y.-C., Tseng, W.-L., Chen, H.-H., & Tseng, Y.-H. (2015). Mining browsing behaviors for objectionable content filtering. *Journal of the Association for Information Science and Technology*, *66*(5), 930-942. https://doi.org/10.1002/asi.23217

Lee, P. Y., Hui, S. C., & Fong, A. C. M. (2003). A structural and content-based analysis for Web filtering. *Internet Research*, *13*(1), 27-37. https://doi.org/10.1108/10662240310458350

Lee, P. Y., Hui, S. C., & Fong, A. C. M. (2005). An intelligent categorization engine for bilingual web content filtering. *IEEE Transactions on Multimedia*, *7*(6), 1183-1190. https://doi.org/10.1109/tmm.2005.858414

Lee, Y., & Cho, J. (2021). Web document classification using topic modeling based document ranking. *Int. J. Electr. Comput. Eng.(2088-8708)*, *11*, 2386-2392.

Leguen-deVarona, I., Madera, J., Martínez-López, Y., & Hernández-Nieto, J. C. (2020). Over-sampling imbalanced datasets using the Covariance Matrix. *EAI Endorsed Transactions on Energy Web*, *7*(27), e2-e2.

Lesniewska-Choquet, C., Mauris, G., Atto, A. M., & Mercier, G. (2019). On elliptical possibility distributions. *IEEE Transactions on Fuzzy Systems*, *28*(8), 1631-1639.

Li, C., Wang, H., Zhang, Z., Sun, A., & Ma, Z. (2016). Topic modeling for short texts with auxiliary word embeddings. Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval,

Li, W., & McCallum, A. (2006). Pachinko allocation: DAG-structured mixture models of topic correlations. Proceedings of the 23rd international conference on Machine learning,

Lin, C., He, Y., Everson, R., & Ruger, S. (2011). Weakly supervised joint sentiment-topic detection from text. *IEEE Transactions on Knowledge and Data Engineering*, *24*(6), 1134-1145.

Linton, M., Teo, E. G. S., Bommes, E., Chen, C., & Härdle, W. K. (2017). Dynamic topic modelling for cryptocurrency community forums. In *Applied Quantitative Finance* (pp. 355-372). Springer.

Liu, S., & Forss, T. (2015a). New classification models for detecting Hate and Violence web content. 2015 7th international joint conference on knowledge discovery, knowledge engineering and knowledge management (IC3K),

Liu, S., & Forss, T. (2015b). *Text Classification Models for Web Content Filtering and Online Safety* 2015 IEEE International Conference on Data Mining Workshop (ICDMW),

Liu, S., & Forss, T. (2015c). Text classification models for web content filtering and online safety. 2015 IEEE international conference on data mining workshop (ICDMW),

Liu, Y., Du, F., Sun, J., Jiang, Y., He, J., Zhu, T., & Sun, C. (2018). A crowdsourcing-based topic model for service matchmaking in Internet of Things. *Future Generation Computer Systems*, *87*, 186-197.

Ma, T., Al-Sabri, R., Zhang, L., Marah, B., & Al-Nabhan, N. (2020). The impact of weighting schemes and stemming process on topic modeling of Arabic long and short texts. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, *19*(6), 1-23.

Maarten Grootendorst. (2022). BERTopic (Version 0.15.0) [Software]. BERTopic: Neural topic modeling with a class-based TF-IDF procedure. *Available at https://maartengr.github.io/BERTopic/index.html*.

MacAvaney, S., Yao, H.-R., Yang, E., Russell, K., Goharian, N., & Frieder, O. (2019). Hate speech detection: Challenges and solutions. *PLoS One*, *14*(8), e0221152.

Mahto, D. K., & Singh, L. (2016). A dive into Web Scraper world. 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom),

Manotas, E. C., & Gonzalez-Perez, M. A. (2020). Internationalization and performance of small and medium-sized enterprises from emerging economies: Using hazards methodology for competitiveness study. *Competitiveness Review: An International Business Journal*.

Martellozzo, E., Monaghan, A., Adler, J. R., Davidson, J., Leyva, R., & Horvath, M. A. (2016). "I wasn't sure it was normal to watch it…" A quantitative and qualitative examination of the impact of online pornography on the values, attitudes, beliefs and behaviours of children and young people. 87.

Martin, G. P., Sperrin, M., Snell, K. I., Buchan, I., & Riley, R. D. (2021). Clinical prediction models to predict the risk of multiple binary outcomes: a comparison of approaches. *Statistics in medicine*, *40*(2), 498-517.

Mcauliffe, J., & Blei, D. (2007). Supervised topic models. *Advances in neural information processing systems*, *20*.

Mcauliffe, J. D., & Blei, D. M. (2008). Supervised topic models. *Advances in neural information processing systems*, 121-128.

McCallum, A.K. (2002). Mallet [Software Toolkit]. MALLET: A Machine Learning for Language Toolkit. *Available at http://mallet.cs.umass.edu/index.php*.

McKinney, W. (2010). Data structures for statistical computing in python. In *Proceedings of the 9th Python in Science Conference* (Vol. 445, pp. 56 - 61). https://doi.org/10.25080/Majora-92bf1922-00a

MCMC. (2017). *Internet users survey 2017*. Malaysian Communications and Multimedia Commission. https://www.mcmc.gov.my/skmmgovmy/media/General/pdf/MCMC-Internet-Users-Survey-2017.pdf

Mimno, D., Wallach, H., Talley, E., Leenders, M., & McCallum, A. (2011). Optimizing semantic coherence in topic models. Proceedings of the 2011 conference on empirical methods in natural language processing,

Mimno, D. M., & McCallum, A. (2008). Topic models conditioned on arbitrary features with Dirichlet-multinomial regression. UAI,

Minh, H. T., Hoang, N. N., Van Hau, B., & Dang, N. (2022). An Ingeniously Integrated Two-Factor Authentication Scheme for Smart Communications and Control Systems. *SSRN 4203668*.

Modi, S. S., & Jagtap, S. B. (2018). Multimodal Web Content Mining to Filter Non-learning Sites Using NLP. International conference on Computer Networks, Big data and IoT,

Moore, S. (2019). *Filtering Network Data Transfers* (U.S. Patent No.

Mulunda, C. K., Wagacha, P. W., & Muchemi, L. (2018). Review of trends in topic modeling techniques, tools, inference algorithms and applications. 2018 5th International Conference on Soft Computing & Machine Intelligence (ISCMI),

Nagulendra, S., & Vassileva, J. (2016). Providing awareness, explanation and control of personalized filtering in a social networking site. *Information Systems Frontiers*, *18*, 145-158.

Nanda, S., Danilak, R., Gyugyi, P. J., Maufer, T. A., Sidenblad, P. J., Jha, A. K., & Rajagopalan, A. (2008). Using TCP/IP offload to accelerate packet filtering. *U.S. Patent No. 7,420,931*.

Nanny, N. (2019). *Internet Filter*. Net Nanny. https://www.netnanny.com/features/internet-filter/

Narayanan, B. K., M, R. B., J, S. M., & M, N. (2018). Adult content filtering: Restricting minor audience from accessing inappropriate internet content. *Education and Information Technologies*, *23*(6), 2719-2735. https://doi.org/10.1007/s10639-018-9738-y

Narayanan, B. K., Moses, S., & Nirmala, M. (2018). Adult content filtering: Restricting minor audience from accessing inappropriate internet content. *Education and Information Technologies*, *23*(6), 2719-2735. https://doi.org/10.1007/s10639-018-9738-y

Narwal, N. (2020). Web page filtering for kids. *International Journal of Information Technology*. https://doi.org/10.1007/s41870-020-00474-0

Narwal, N., & Sharma, S. K. (2016). Web informative content identification and filtering using machine learning technique. *International Journal of Data Analysis Techniques and Strategies*, *8*(4), 332-347.

Netcraft. (2018). February 2018 Web Server Survey. https://news.netcraft.com/archives/2018/02/13/february-2018-web-server-survey.html

Netcraft. (2018, 19 January). January 2018 Web Server Survey. https://news.netcraft.com/archives/2018/01/19/january-2018-web-server-survey.html

Newman, D., Noh, Y., Talley, E., Karimi, S., & Baldwin, T. (2010). Evaluating topic models for digital libraries. Proceedings of the 10th annual joint conference on Digital libraries,

O'callaghan, D., Greene, D., Carthy, J., & Cunningham, P. (2015). An analysis of the coherence of descriptors in topic modeling. *Expert Systems with Applications*, *42*(13), 5645-5657.

Padilla-Walker, L. M., Coyne, S. M., Fraser, A. M., Dyer, W. J., & Yorgason, J. B. (2012). Parents and adolescents growing up in the digital age: latent growth curve analysis of proactive media monitoring. *J Adolesc*, *35*(5), 1153-1165. https://doi.org/10.1016/j.adolescence.2012.03.005

Page, L., Brin, S., Motwani, R., & Winograd, T. (1999). *The PageRank citation ranking: Bringing order to the web*.

Patel, O., Tiwari, A., Patel, V., & Gupta, O. (2015). Quantum based neural network classifier and its application for firewall to detect malicious web request. 2015 IEEE symposium series on computational intelligence,

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of machine Learning research*, *12*, 2825--2830.

Phadia, E. G., & Phadia, E. G. (2016). Prior Processes: An Overview. *Prior Processes and Their Applications: Nonparametric Bayesian Estimation*, 1-17.

Porouhan, P., & Premchaiswadi, W. (2017). Process Mining and Learners' Behavior Analytics in a Collaborative and Web-Based Multi-Tabletop Environment. *International Journal of Online Pedagogy and Course Design (IJOPCD)*, *7*(3), 29-53.

Pratiba, D., Abhay, M., Dua, A., Shanbhag, G. K., Bhandari, N., & SINGH, U. (2018). Web scraping and data acquisition using Google scholar. 2018 3rd International Conference on Computational Systems and Information Technology for Sustainable Solutions (CSITSS),

Priyadharshini, K., Kavindra, J., Kalaivaani, K., Lakshana, S., & Mrudhhula, V. (2023). Identification and Selection of Random Forest Algorithm for Predicting Hypothyroid. 2023 9th International Conference on Advanced Computing and Communication Systems (ICACCS),

Qustodio. (2019). *Keep your child's online experience safe, fun, and productive*. qustodio. https://www.qustodio.com/en/premium-special-promo-b/?utm_source=google&utm_medium=cpc&utm_campaign=adw_ww_web_brand___brand_ww&utm_term=brand&utm_content=&gclid=EAIaIQobChMI39zkhJ304QIViYyPCh36OgKMEAAYASAAEgLVmvD_BwE

Rajalakshmi, R., Tiwari, H., Patel, J., Kumar, A., & Karthik, R. (2020). Design of Kids-specific URL Classifier using Recurrent Convolutional Neural Network. *Procedia Computer Science*, *167*, 2124-2131.

Rama, T. (2016). Chinese Restaurant Process for cognate clustering: A threshold free approach. *arXiv preprint arXiv:1610.06053*.

Rao, R. S., Vaishnavi, T., & Pais, A. R. (2020). CatchPhish: detection of phishing websites by inspecting URLs. *Journal of Ambient Intelligence and Humanized Computing*, *11*(2), 813-825.

Rehurek, R. a. S., Petr. (2011). Gensim (Version 4.3.2) [Software]. Gensim--python framework for vector space modelling. *NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic*, *3*(2). https://pypi.org/project/gensim/ , urldate = 2022-04-28

Research, Z. M. (2018). *Global Parental Control Market Will Reach USD 3,300 million by 2025: Zion Market Research*.

Roberts, D. A. (2021). Stochaskell: a common platform for probabilistic programming research and applications.

Röder, M., Both, A., & Hinneburg, A. (2015). Exploring the space of topic coherence measures. Proceedings of the eighth ACM international conference on Web search and data mining,

Ruan, S., & Stormo, G. D. (2017). Inherent limitations of probabilistic models for protein-DNA binding specificity. *PLoS computational biology*, *13*(7), e1005638.

Sahingoz, O. K., Buber, E., Demir, O., & Diri, B. (2019). Machine learning based phishing detection from URLs. *Expert Systems with Applications*, *117*, 345-357.

Sayadi, K., Bui, Q. V., & Bui, M. (2015). Multilayer classification of web pages using random forest and semi-supervised latent dirichlet allocation. 2015 15th International Conference on Innovations for Community Services (I4CS),

Selenium for Python. https://pypi.org/project/selenium , urldate = 2022-04-28

Shyry, S. P., & Jinila, Y. B. (2021). Detection and prevention of spam mail with semantics-based text classification of collaborative and content filtering. Journal of Physics: Conference Series,

Sievert, C., & Shirley, K. (2015). pyLDAvis (Version 3.4.1) [Library]. A visualization tool for interpreting and understanding topics from topic models. Available at [URL]

Singh, A. (2020). Malicious and benign webpages dataset. *Data in Brief*, *32*, 106304.

Soup, B. (2020). beautifulsoup4 (Version 4.12.2) [Library]. *Beautiful Soup Documentation*. Beautiful Soup. Available at https://beautiful-soup-4.readthedocs.io/en/latest/.

Stanford University. (2009). Stanford Topic Modeling Toolbox (TMT) (Version 0.2) [Software]. *Available at https://downloads.cs.stanford.edu/nlp/software/tmt/tmt-0.2*.

Steyvers, M., Smyth, P., Rosen-Zvi, M., & Griffiths, T. (2004). Probabilistic author-topic models for information discovery. Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining,

Stroud, J. C., Lu, Z., Sun, C., Deng, J., Sukthankar, R., Schmid, C., & Ross, D. A. (2020). Learning video representations from textual web supervision. *arXiv preprint arXiv:2007.14937*.

Syed, S., & Spruit, M. (2017). *Full-Text or Abstract? Examining Topic Coherence Scores Using Latent Dirichlet Allocation* 2017 IEEE International Conference on Data Science and Advanced Analytics (DSAA),

Tang, J., Jin, R., & Zhang, J. (2008). A topic modeling approach and its integration into the random walk framework for academic search. 2008 Eighth IEEE International Conference on Data Mining,

Thivaharan, S., Srivatsun, G., & Sarathambekai, S. (2020). A survey on python libraries used for social media content scraping. 2020 International Conference on Smart Electronics and Communication (ICOSEC),

Tippett, N., & Wolke, D. (2014). Socioeconomic status and bullying: a meta-analysis. *Am J Public Health*, *104*(6), e48-59. https://doi.org/10.2105/AJPH.2014.301960

Tld Library. https://pypi.org/project/tld , urldate = 2022-04-28

Top, N. (2016). Socio-Demographic Differences in Parental Monitoring of Children in Late Childhood and Adolescents' Screen-Based Media Use. *Journal of Broadcasting & Electronic Media*, *60*(2), 195-212. https://doi.org/10.1080/08838151.2016.1164168

Tyagi, N., & Gupta, S. K. (2018). Web structure mining algorithms: A survey. Big Data Analytics: Proceedings of CSI 2015,

UNICEF Malaysia & Digi. (2015). Talk to your children about the internet. https://www.unicef.org/malaysia/reports/talk-your-children-about-internet

Valkenburg, P. M., Krcmar, M., Peeters, A. L., & Marseille, N. M. (1999). Developing a scale to assess three styles of television mediation:"Instructive mediation,""restrictive mediation," and "social coviewing". *Journal of Broadcasting & Electronic Media*, *43*(1), 52-66.

Varadharajan, V. (2010). Internet filtering-issues and challenges. *IEEE Security & Privacy*, *8*(4), 62-65.

Vayansky, I., & Kumar, S. A. (2020). A review of topic modeling methods. *Information Systems*, *94*, 101582.

Vrbančič, G., Fister Jr, I., & Podgorelec, V. (2020). Datasets for phishing websites detection. *Data in Brief*, *33*, 106438.

W3C Recommendation. (2009, 1 September). Protocol for Web Description Resources (POWDER): Description Resources. https://www.w3.org/TR/powder-dr/

Wan, J., Liu, M., Yi, J., & Zhang, X. (2015). Detecting spam webpages through topic and semantics analysis. 2015 Global Summit on Computer & Information Technology (GSCIT),

Wang, C., Blei, D., & Heckerman, D. (2012). Continuous time dynamic topic models. *arXiv preprint arXiv:1206.3298*.

Wang, J., He, K., & Yang, M. (2020). Topic discovery by spectral decomposition and clustering with coordinated global and local contexts. *International Journal of Machine Learning and Cybernetics*, 2475-2487.

Wong, Y. C. (2010). Cyber-Parenting: Internet Benefits, Risks and Parenting Issues. *Journal of Technology in Human Services*, *28*(4), 252-273. https://doi.org/10.1080/15228835.2011.562629

Wu, I., & Hwang, W.-H. (2013). A genre-based fuzzy inference approach for effective filtering of movies. *Intelligent Data Analysis*, *17*(6), 1093-1113.

WZB Berlin Social Science Center. (2019). Tmtoolkit (Version 0.11.2) [Software]. *Available at https://github.com/WZBSocialScienceCenter/tmtoolkit*.

Xing, W., & Ghorbani, A. (2004). Weighted pagerank algorithm. Proceedings. Second Annual Conference on Communication Networks and Services Research, 2004.,

Xu, S., Shi, Q., Qiao, X., Zhu, L., Jung, H., Lee, S., & Choi, S.-P. (2014). Author-Topic over Time (AToT): a dynamic users' interest model. In *Mobile, ubiquitous, and intelligent computing* (pp. 239-245). Springer.

Yang, Y., Liu, Y., Lu, X., Xu, J., & Wang, F. (2020). A named entity topic model for news popularity prediction. *Knowledge-Based Systems*, *208*, 106430.

Yenala, H., Jhanwar, A., Chinnakotla, M. K., & Goyal, J. (2018). Deep learning for detecting inappropriate content in text. *International Journal of Data Science and Analytics*, *6*(4), 273-286.

Zeng, J., Duan, J., & Wu, C. (2013). *Adaptive Topic Modeling for Detection Objectionable Text* 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT),

Zeniarja, J., Sani, R. R., Luthfiarta, A., Susanto, H. A., Hidayat, E. Y., Salam, A., & Mahendra, L. I. B. (2018, September). *Search Engine for Kids with Document Filtering and Ranking Using Naive Bayes Classifier* International Seminar on Application for Technology of Information and Communication,

Zhang, A., Zhu, J., & Zhang, B. (2013). Sparse relational topic models for document networks. Joint European Conference on Machine Learning and Knowledge Discovery in Databases,

Zhao, C., Zhang, Y., Zang, T., Liang, Z., & Wang, Y. (2018). A Stacking Approach to Objectionable-Related Domain Names Identification by Passive DNS Traffic (Short Paper). International Conference on Collaborative Computing: Networking, Applications and Worksharing,

Zhao, J., Huang, J. X., Deng, H., Chang, Y., & Xia, L. (2021). Are topics interesting or not? An LDA-based topic-graph probabilistic model for web search personalization. *ACM Transactions on Information Systems (TOIS)*, *40*(3), 1-24.

Zhao, W. X., Jiang, J., Weng, J., He, J., Lim, E.-P., Yan, H., & Li, X. (2011). Comparing twitter and traditional media using topic models. Advances in Information Retrieval: 33rd European Conference on IR Research, ECIR 2011, Dublin, Ireland, April 18-21, 2011. Proceedings 33,

Zhu, J., Zhu, M., Wang, H., & Tsou, B. K. (2009). Aspect-based sentence segmentation for sentiment summarization. Proceedings of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion,

Zuo, Y., Wu, J., Zhang, H., Lin, H., Wang, F., Xu, K., & Xiong, H. (2016). Topic modeling of short texts: A pseudo-document view. Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining,