UTILIZING BIG DATA THROUGH TEXT ANALYSIS AND VISUALIZATION FRAMEWORK:
A CASE STUDY OF ROAD CONSTRUCTION BUDGETING

CHANWIT KONGTHONG

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE
DEGREE OF MASTER OF SCIENCE IN COMPUTER SCIENCE
DEPARTMENT OF COMPUTER SCIENCE SCHOOL OF SCIENCE
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
2024
KMITL-2024-SC-M-002-046

Thesis Title	Utilizing Big Data Through Text Analysis And Visualization Framework:A Case Study of Road Construction Budgeting
Student Name	Chanwit Kongthong
Student ID	61605056
Degree	Master of Science (Computer Science)
Department	Computer Science
Year	2024
Thesis Advisor	Asst. Prof. Dr. Sarun Intakosum

Abstract

Processing Thai language texts can be a challenge due to the complexities of the language, particularly texts from social media and online platforms. This study introduces an analysis and visualization framework specifically designed to tackle the intricacies associated with processing the Thai language data within the context of online textual content, by utilizing natural language processing (NLP) and visualization techniques. The objectives of this study were to develop an effective Thai text data analysis and visualization framework that allows us to effectively and automatically get a better understanding of the content embedded in Thai textual data. The methodology initiated with a review of existing analysis frameworks and visualization techniques with a specific focus on Thai. The proposed Thai analysis and visualization framework consists of multiple stages. Each stage is tailored to accommodate the intricacies of the Thai language, facilitating improved information extraction and text comprehension. The proposed visualization techniques utilize interactive graphs, such as bar charts, line charts, pie charts and donut charts, to offer intuitive and insightful representations of the processed data.

Keywords: data visualization, natural language processing, text mining

Acknowledgements

I would like to express my deepest gratitude to those who have supported and guided me throughout the course of this research.

Firstly, I extend my heartfelt thanks to my first advisor, Teacher Kulsawad, whose invaluable guidance, support, and encouragement were crucial to the completion of this thesis. Your insightful feedback and unwavering support have been instrumental in shaping this work, and I am immensely grateful for the time and effort you invested in my research.

I would also like to sincerely thank my second advisor, Teacher Sarun, for your expert advice and constructive feedback. Your perspectives and suggestions have significantly enhanced the quality of this research, and your support has been greatly appreciated. Thank you both for your dedication and for believing in my work. Your mentorship has been a vital part of my academic journey, and I am deeply thankful for your contributions.

Chanwit Kongthong

Table of contents

Table of contents

Table of contents

List of tables

List of figures

List of figures

Chapter 1

Introduction

1.1 Research Motivation

Processing Thai language texts is difficult because of the language's complexities, especially when dealing with texts from social media and online platforms. This paper introduces a framework designed to address these challenges by using natural language processing (NLP) and visualization techniques to better understand Thai language data from online content. In today's digital age, vast amounts of text data are generated across the internet. This textual data, encompassing both structured and unstructured formats, holds significant potential for extracting valuable insights. However, the sheer volume and variety of this data pose substantial challenges in terms of processing and analysis. Simple, effective visualizations are essential to transform this raw data into meaningful insights that can be easily interpreted and used for further applications.

In this research, we focus on extracting and analyzing data related to government budgeting. Government budgeting is a critical domain where transparency and accessibility of information can significantly impact public understanding and trust. Various news websites publish daily updates on government activities, including detailed reports on budgeting and spending on ongoing projects. Additionally, official Thai government budgeting websites provide substantial information. However, our research identified limitations in the depth of insights these sources offer and the functionalities available for data analysis.

In exploring tools and methodologies from previous studies, we found that most research focuses on English language processing. There is a substantial body of work and numerous methods available for handling English text. In contrast, there is significantly less research and fewer methodologies specifically tailored for Thai text processing. This gap highlights the need for specialized approaches to effectively manage and analyze Thai language data.

With a correct and proper way to handle Thai text processing, we believe that Thai text data can be used to achieve many useful outcomes in both the business and educational sectors. For businesses, improved Thai text processing can lead to better customer insights, enhanced decision-making, and more effective marketing strategies. For example, in Thailand, some businesses already use Thai text processing to enhance their operations. Banks use sentiment analysis to better understand customer feelings and trends, enabling them to launch more effective promotions. Additionally, chatbots powered by Thai language processing improve customer service, and monitoring social media helps businesses stay aware of current interests and trends.

In summary, this research seeks to leverage advanced NLP and visualization techniques to overcome the challenges of processing Thai language texts, particularly in the context of government budgeting. By enhancing the extraction and visualization of data, we aim to provide deeper insights and a useful framework for both seasoned researchers and newcomers in the field.

1.2 Objectives of the Study

The primary aim of this study is to develop an effective framework for analyzing and visualizing Thai text data. This objective is driven by the need to overcome the inherent complexities associated with the Thai language and to provide meaningful insights from vast amounts of unstructured and structured textual data. To achieve this overarching goal, the study is guided by the following specific objectives:

1) To Create a System for Analyzing Thai Text Data

2) To Use NLP Techniques to Handle and Understand the Complexities of the Thai Language

3) To Develop Visual Tools that Make the Processed Data Easy to Understand

1.3 Scope(s) of the Study

The scope of this study encompasses several key areas essential for the development of an effective framework for analyzing and visualizing Thai text data. The study is structured to ensure a comprehensive approach to addressing the

complexities of Thai language processing, focusing on specific domains and utilizing diverse data sources, considering some areas are complex and challenging to collect data from. Recognize the challenges in data collection from certain sources due to complexity and accessibility issues. The scope includes the following components:

1) Review of Existing Analysis Frameworks and Visualization Techniques.

2) Data Collection from https://govspending.data.go.th/ , unstructured data from news website and pdf file.

3) Preprocessing of Collected Data using PyThaiNLP (python library) and regular expression.

4) Focus on Government Budgeting Domain – Road Construction.

5) Visualization of Budgeting Data using tool from JavaScript: chart.js

By delineating these specific scopes, the study ensures a structured and focused approach to developing and validating the Thai text analysis and visualization framework.

1.4 Benefits of the Study

The study aims to deliver significant benefits through the development and implementation of a specialized framework for analyzing and visualizing Thai text data. The proposed framework addresses the unique complexities of the Thai language, providing a comprehensive solution for extracting and interpreting valuable information.

Chapter 2

Theory and literature reviews

In this chapter, we delve into an extensive exploration of the existing literature and theoretical frameworks relevant to Thai text processing, setting the stage for our study. The primary objective is to comprehensively review prior research and studies conducted in the field, shedding light on the approaches, methodologies, tools, and techniques utilized in handling Thai text data.

2.1 Importance of Reviewing Existing Literature and Theoretical Frameworks

In this section we study by examining existing literature, we gain valuable insights into how Thai text processing is approached and managed. This involves understanding the methodologies employed, the challenges encountered, and the strategies devised to overcome them. The review provides an in-depth understanding of the various approaches and methodologies adopted by researchers in processing Thai text. This includes techniques such as natural language processing (NLP), sentiment analysis, and text mining, among others. Through the review, we gain clarity on the workflow and processes involved in Thai text processing. This encompasses stages such as data collection, preprocessing, analysis, and visualization, allowing us to discern best practices and potential areas for improvement. Reviewing existing literature and frameworks saves valuable time by providing a wealth of information and insights accumulated by previous researchers. This enables us to build upon established knowledge and avoid reinventing the wheel, streamlining the research process and enhancing the effectiveness of our study.

2.2 Importance of Text Processing Tools

Text processing tools are fundamental components in the language processing pipeline, providing the necessary functionality to transform raw text data into a structured format suitable for analysis. These tools enable researchers and practitioners to perform a variety of crucial tasks that lay the groundwork for more advanced natural language processing (NLP) operations. The key aspects of text processing tools include text cleaning, tokenization, and feature extraction, each playing a vital role in preparing textual data for further analysis.

2.2.1 Text Cleaning

Text cleaning is the first and most crucial step in text processing. Raw textual data often contains noise such as punctuation, special characters, HTML tags, and other irrelevant elements that can obscure meaningful information. Text cleaning tools help in:

1) Removing Unwanted Characters: By stripping out unnecessary punctuation marks, special symbols, and formatting artifacts, these tools ensure that the text is in a clean and standardized format.
2) Handling Inconsistencies: Addressing issues such as inconsistent casing (e.g., upper and lower case), misspellings, and irregular whitespace can significantly improve the quality of the data.
3) Normalizing Text: Converting text to a consistent format, such as lowercasing all letters or converting numbers to words, helps in maintaining uniformity across the dataset.

2.2.2 Tokenization

Tokenization is the process of breaking down text into smaller units called tokens, which can be words, phrases, or even entire sentences. This step is essential for several reasons:

1) Facilitating Analysis: By converting text into manageable chunks, tokenization makes it easier to perform subsequent analyses

such as frequency counting, part-of-speech tagging, and syntactic parsing.

2) Context Understanding: Proper tokenization helps in preserving the context of words within a sentence, which is crucial for tasks like sentiment analysis and machine translation.
3) Handling Multi-word Expressions: In languages like Thai, where words are not always separated by spaces, tokenization tools can identify and segment multi-word expressions accurately.

2.2.3 Feature Extraction

Feature extraction involves transforming text data into numerical representations that can be used by machine learning algorithms. This process is critical for tasks such as text classification, clustering, and sentiment analysis. Key aspects include:

1) Vectorization: Converting text into numerical vectors using techniques like Term Frequency-Inverse Document Frequency (TF-IDF), word embeddings (e.g., Word2Vec, GloVe), or more advanced methods like BERT and GPT.
2) Dimensionality Reduction: Reducing the number of features while preserving the essential information helps in improving computational efficiency and model performance.
3) Selection of Relevant Features: Identifying and selecting the most informative features ensures that the models focus on the most significant aspects of the text, improving their accuracy and interpretability.

2.3 Regular Expressions (regex)

Regular expressions, often abbreviated as regex, stand as indispensable tools in the realm of computational linguistics and text processing. These versatile patterns enable users to define complex search patterns, facilitating efficient extraction, substitution, and manipulation of textual data. In the context of Thai text processing, regex plays a pivotal role in various tasks, ranging from pattern matching to text cleaning and tokenization.

2.3.1 Pattern Matching

Regex can identify specific sequences of characters within text, such as email addresses, phone numbers, or dates. For instance:

1) Email Addresses: The regex pattern [a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,} can be used to find email addresses within a body of text.
2) Phone Numbers: To identify phone numbers, a pattern like \b\d{3}[-.]?\d{3}[-.]?\d{4}\b can match various formats (e.g., 123-456-7890, 123.456.7890).
3) Dates: A regex such as \b\d{2}[\/-]\d{2}[\/-]\d{4}\b can match dates in formats like 12/31/2023 or 31-12-2023.

2.3.2 Text Cleaning

Regex is extensively used for removing unwanted characters, punctuation, and other noise from text. For example:

1) Removing HTML Tags: The regex pattern <[^>]+> can be used to strip HTML tags from text.
2) Removing Punctuation: To remove all punctuation marks, a pattern like [^\w\s] can match and remove characters that are not word characters or whitespace.
3) Whitespace Normalization: A pattern such as \s+ can replace multiple spaces with a single space, ensuring consistent spacing within the text.

2.3.3 Tokenization

Regex patterns can specify delimiters that segment text into tokens, facilitating further analysis. For instance:

1) Word Tokenization: Using a pattern like \b\w+\b, regex can split text into words based on word boundaries.
2) Sentence Tokenization: A regex pattern such as [^.!?]+[.!?] can be used to split text into sentences based on sentence-ending punctuation marks.
3) Custom Delimiters: In scenarios where text contains custom delimiters, such as commas or semicolons, a pattern like [^,;]+ can be used to tokenize the text accordingly.

2.4 Natural Language Processing (NLP)

Natural Language Processing (NLP) is a field at the intersection of computer science, artificial intelligence, and linguistics that focuses on the interaction between computers and human (natural) languages. The goal of NLP is to enable computers to understand, interpret, and generate human language in a way that is both meaningful and useful. NLP involves the use of computational techniques to process and analyze large amounts of natural language data. The main objectives are to enable machines to read, understand, and derive meaning from human languages, and to facilitate human-computer interaction. There are the core components of NLP:

2.4.1Text Preprocessing

The initial step in NLP where raw text is cleaned and prepared for further analysis. This includes tasks like tokenization (splitting text into words or sentences), removing stop words (common words like 'the', 'is', etc.), and stemming or lemmatization (reducing words to their root forms).

2.4.2 Syntax and Parsing

Understanding the grammatical structure of sentences. This involves parsing sentences to identify parts of speech (nouns, verbs, adjectives, etc.) and their relationships.

2.4.3 Semantic Analysis

Extracting meaning from text. This includes named entity recognition (identifying proper names, dates, locations, etc.), sentiment analysis (determining the emotional tone), and semantic role labeling (identifying the role played by each entity in a sentence).

2.4.4 Machine Translation

Translating text from one language to another while maintaining its meaning and context.

2.4.5 Information Retrieval and Extraction

Finding relevant information within large datasets and extracting structured data from unstructured text.

2.5 PyThaiNLP

PyThaiNLP is a Python library specifically designed for natural language processing tasks involving the Thai language. Developed by the Data Science Laboratory at the National Electronics and Computer Technology Center (NECTEC), PyThaiNLP offers a comprehensive suite of functionalities tailored to the intricacies of Thai text processing such as:

2.5.1 Word Segmentation

One of the fundamental tasks in Thai language processing is word segmentation, where text is divided into individual tokens or words. PyThaiNLP provides robust algorithms for word segmentation, allowing researchers to break down raw Thai text into meaningful units for subsequent analyses. For example, consider a scenario where a researcher aims to analyze sentiment in Thai social media posts. By leveraging PyThaiNLP's word segmentation

capabilities, researchers can tokenize the text, enabling sentiment analysis algorithms to operate at the word level and capture nuances in language use.

```
from pythainlp.tokenize import word_tokenize

text = "โอเคบ่พวกเรารักภาษาบ้านเกิด"

word_tokenize(text, engine="newmm")
# output: ['โอเค', 'บ่', 'พวกเรา', 'รัก', 'ภาษา', 'บ้านเกิด']

word_tokenize(text, engine='attacut')
# output: ['โอเค', 'บ่', 'พวกเรา', 'รัก', 'ภาษา', 'บ้านเกิด']
```

Figure 2.1 PyThaiNLP word Tokenization

2.5.2 Part-of-Speech Tagging

PyThaiNLP offers pre-trained models and algorithms for part-of-speech tagging, a task crucial for understanding the grammatical structure of Thai sentences. Part-of-speech tagging assigns grammatical categories, such as nouns, verbs, adjectives, and adverbs, to words in Thai text. This functionality enables researchers to perform syntactic analysis, semantic role labeling, and information extraction tasks with precision and accuracy. For instance, in educational settings, PyThaiNLP's part-of-speech tagging capabilities can aid language learners in identifying and understanding the roles of words within sentences, facilitating comprehension and language acquisition.

The following table shows Universal POS tags as used in Universal Dependencies (UD):

Abbreviation	Part-of-Speech tag	Examples
ADJ	Adjective	ใหม่, พิเศษ , ก่อน, มาก, สูง
ADP	Adposition	แม้, ว่า, เมื่อ, ของ, สำหรับ
ADV	Adverb	ก่อน, ก็, เล็กน้อย, เลย, สุด
AUX	Auxiliary	เป็น, ใช่, คือ, คล้าย
CCONJ	Coordinating conjunction	แต่, และ, หรือ
DET	Determiner	ที่, นี้, ซึ่ง, ทั้ง, ทุก, หลาย
INTJ	Interjection	อุ้ย, โอ้ย
NOUN	Noun	ตำมือ, พวก, สนาม, ก็ฟ้า, บัญชี
NUM	Numeral	5,000, 103.7, 2004, หนึ่ง, ร้อย
PART	Particle	มา ขึ้น ไม่ ได้ เข้า
PRON	Pronoun	เรา, เขา, ตัวเอง, ใคร, เธอ
PROPN	Proper noun	โอบามา, แคปิตอลฮิล, จิโอพี, ไมเคิล
PUNCT	Punctuation	(,), “, ...
SCONJ	Subordinating conjunction	หาก
VERB	Verb	เปิด, ให้, ใช้, เผชิญ, อ่าน

Figure 2.2 PyThaiNLP word tag

2.5.3 Sentiment Analysis

PyThaiNLP offers tools and resources for sentiment analysis in Thai text, allowing researchers to analyze the sentiment or emotional tone conveyed by textual data. This functionality is particularly valuable for applications such as social media monitoring, customer feedback analysis, and opinion mining.

2.5.4 Named Entity Recognition

The library includes capabilities for named entity recognition (NER), identifying and classifying entities such as names of people, organizations,

locations, and dates within Thai text. This feature is essential for information extraction tasks and knowledge discovery from textual data.

2.5.5 Text Summarization

This function summarizes text based on frequency of words. First tokenize sentence from the given text with pythainlp.tokenize.sent_tokenize(). Then, computes frequencies of tokenized words (with pythainlp.tokenize.word_tokenize()) in all sentences and normalized with maximum word frequency. The words with normalized frequency that are less than 0.1 or greater than 0.9 will be filtered out from frequency dictionary. Finally, it picks n sentences with highest sum of normalized frequency from all words in the sentence and also appear in the frequency dictionary. Example in figure 2.3

2.5.6 Count word frequency

This function given a list of Thai words with an option to exclude stop words. Example in figure 2.4

2.5.7 Word vector similarity

This function finds the top-10 words that are most similar with respect to from two lists of words labeled as positive and negative. Example in figure 2.5

2..6 Tools and Techniques for Textual Data in Thai

Kongthon proposed the ThaiReSearch framework in fig 2.6, which integrates techniques in information retrieval, database management systems, and data mining to provide useful information to users. This system consolidates research-related information in Thailand from various databases and offers both search and intelligent information analysis functions. It employs statistical analysis, natural language processing, and text mining to support R&D management. Thai R&D managers and

policy planners can use this system to enhance strategic decision-making processes, contributing to a sustainable economy.

```
from pythainlp.summarize import summarize

text = '''
        ท่าเนียบท่าช้าง หรือ วังถนนพระอาทิตย์
        ตั้งอยู่บนถนนพระอาทิตย์ เขตพระนคร กรุงเทพมหานคร
        เดิมเป็นบ้านของเจ้าพระยามหาโยธา (ทอเรียะ คชเสนี)
        บุตรเจ้าพระยามหาโยธานราธิบดีศรีพิชัยณรงค์ (พญาเจ่ง)
        ต้นสกุลคชเสนี เชื้อสายมอญ เจ้าพระยามหาโยธา (ทอเรีย)
        เป็นปู่ของเจ้าจอมมารดากลิ่นในพระบาทสมเด็จพระจอมเกล้าเจ้าอยู่หัว
        และเป็นมรดกตกทอดมาถึง พระเจ้าบรมวงศ์เธอ กรมพระนเรศรวรฤทธิ์
        (พระองค์เจ้ากฤดาภินิหาร)
        ต่อมาในรัชสมัยพระบาทสมเด็จพระจุลจอมเกล้าเจ้าอยู่หัวโปรดเกล้าฯ
        ให้สร้างตำหนัก 2 ชั้น
        เป็นที่ประทับของพระเจ้าบรมวงศ์เธอ
        กรมพระนเรศวรฤทธิ์และเจ้าจอมมารดา
        ต่อมาเรียกอาคารหลักนี้ว่า ตำหนักเดิม
    '''

summarize(text, n=1)
# output: ['บุตรเจ้าพระยามหาโยธานราธิบดีศรีพิชัยณรงค์']

summarize(text, n=3)
# output: ['บุตรเจ้าพระยามหาโยธานราธิบดีศรีพิชัยณรงค์',
# 'เดิมเป็นบ้านของเจ้าพระยามหาโยธา',
# 'เจ้าพระยามหาโยธา']

summarize(text, engine="mt5-small")
# output: ['<extra_id_0> ท่าช้าง หรือ วังถนนพระอาทิตย์
# เขตพระนคร กรุงเทพมหานคร ฯลฯ ดังนี้:
# ที่อยู่ - ศิลปวัฒนธรรม']

text = "ถ้าพูดถึงขนมหวานในตำนานที่ชื่นใจที่สุดแล้วละก็ต้องไม่พ้น น้ำแข็งใส แน่ๆ เพราะว่าเป็นอะไรที่ชื่นใจสุดๆ"
summarize(text, engine="mt5-cpe-kmutt-thai-sentence-sum")
# output: ['น้ำแข็งใสเป็นอะไรที่ชื่นใจที่สุด']
```

Figure 2.3 PyThaiNLP word Summarization

Another notable work is by Katchapakirin, which focuses on detecting words and sentences in social media to identify individuals potentially experiencing depression. This involves processing Thai texts, assigning weights to each word to quantify and measure the degree of depression. However, these works lack essential visualization components that could enhance data interpretation and user engagement.

```python
from pythainlp.util import rank

words = ["บันทึก", "เหตุการณ์", " ", "มี", "การ", "บันทึก", \
"เป็น", " ", "ลายลักษณ์อักษร"]

rank(words)
# output:
# Counter(
#     {
#         ' ': 2,
#         'การ': 1,
#         'บันทึก': 2,
#         'มี': 1,
#         'ลายลักษณ์อักษร': 1,
#         'เป็น': 1,
#         'เหตุการณ์': 1
#     })
```

Figure 2.4 PyThaiNLP word rank function

Find the **top-10** most similar words to the word: "แม่น้ำ".

```
>>> from pythainlp.word_vector import WordVector
>>>
>>> wv = WordVector()
>>> list_positive = ['แม่น้ำ']
>>> list_negative = []
>>> wv.most_similar_cosmul(list_positive, list_negative)
[('ลำน้ำ', 0.8206598162651062), ('ทะเลสาบ', 0.775945782661438),
('ลุ่มน้ำ', 0.7490593194961548), ('คลอง', 0.7471904754638672),
('ปากแม่น้ำ', 0.7354257106781006), ('ฝั่งแม่น้ำ', 0.7120099067687988),
('ทะเล', 0.7030453681945801), ('ริมแม่น้ำ', 0.7015200257301331),
('แหล่งน้ำ', 0.6997432112693787), ('ภูเขา', 0.6960948705673218)]
```

Figure 2.5 PyThaiNLP wordVector function

One of the main problems in searching Thai-language documents is that person or organization names which are transliterated from English to Thai, can be written differently. Therefore, if users spell the name differently in the search query, they might not be able to find information they need. Hence, searching based on phonetic similarity can alleviate such problem. For example, various transliterations for “Smith”, i.e., “สมิ ต,” “สมิ ท,” “สมิ ทธ์ิ ,” result in the same phonetic representation. This representation is then stored in phoneme-based index. When the user search by the name “Smith” with a different spelling – “สมิ ธ,” the query is transformed into phonetic representation before it is used to match with the same representation in the phoneme-based index. As a result, all the relevant items will be returned to the user.

In conclusion, this paper presents a system that integrates research related information in Thailand from various databases. Not only the system offers the way to search and retrieve information, it also provides an intelligent information analysis function using statistical analysis, natural language processing, and text mining. They describe how ThaiReSearch can help support R&D management in several ways. By using the system, the Thai R&D managers and policy planners could improve their strategic decision-making processes towards a sustainable economy.

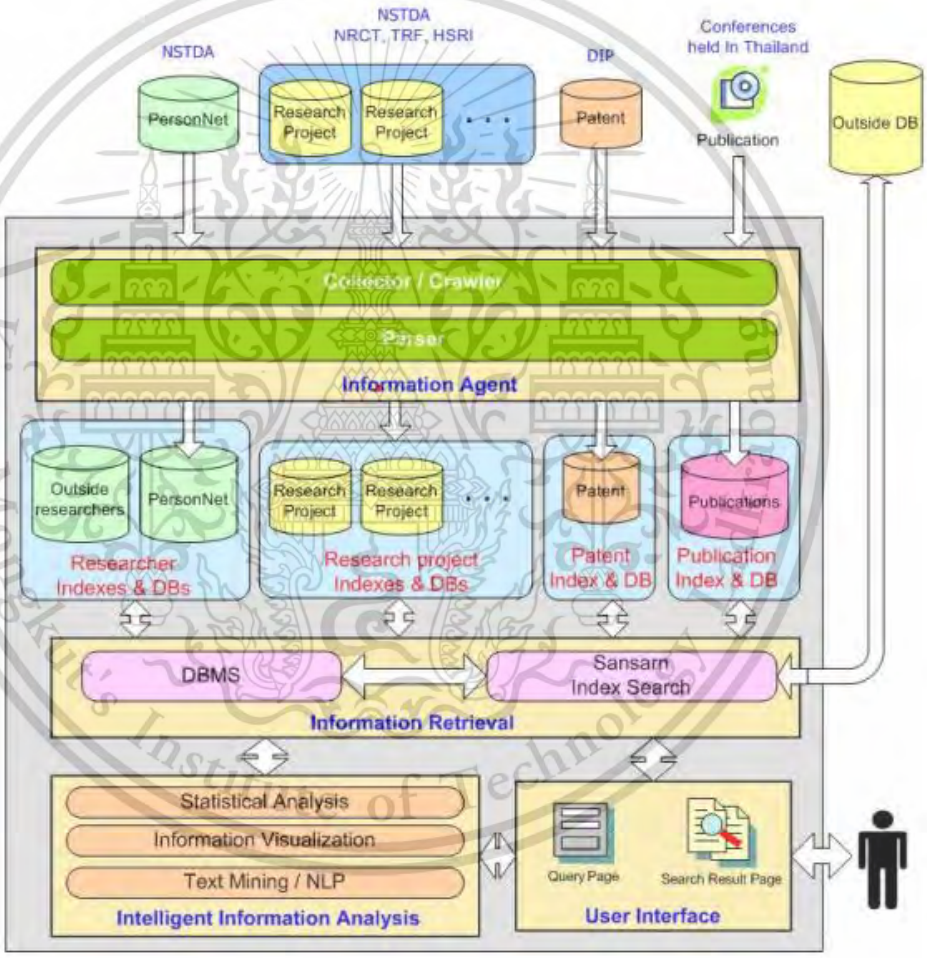

Figure 2.6 ThaiReSearch Framework

2.7 Thai Language Sentiment Analysis Frameworks and Applications

Masdisornchote proposed a sentiment analysis framework for Thai language opinion mining, consisting of three modules: a knowledge corpus construction module, a data preprocessing module, and a pattern analysis module. This framework provides valuable insights into opinion mining (or sentiment analysis), which can be used in various contexts.

For example, political institutions can gain insights into political leanings and influential factors, consumers can make informed purchasing decisions, and entrepreneurs can assess the reputations of their products.

All reviews are collected from Siamphone website which is biggest in providing information about mobile devices. People can share opinions or express their feelings through this website. their work only considers sentences, which contain resources. The total number of reviews is approximately 1,090 sentences. By using their proposed framework, they could systematically classify their sentiments

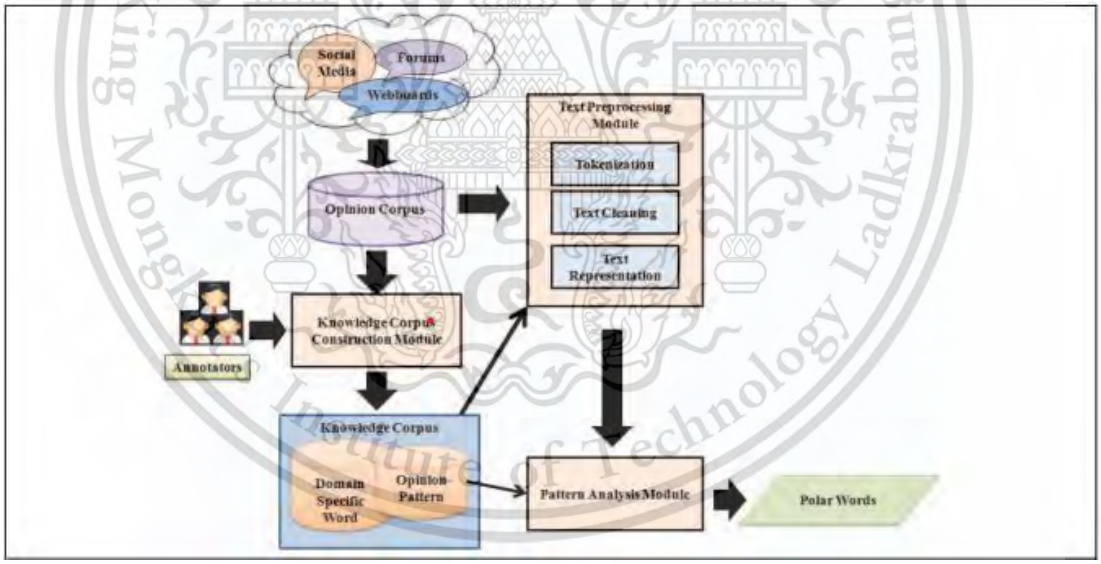

Figure 2.7 Sentiment analysis framework in implicit opinions for Thai language

In conclusion, this paper proposes a framework for sentiment analysis in implicit opinions for Thai language. The framework consists of three modules: Knowledge Corpus Construction Module, Data Preprocessing Module and Pattern Analysis Module. We evaluate the framework within a mobile device domain. Our

experimental results indicate that the proposed framework perform effectively. Regarding our future research, we plan to extend this work to consider opinion expiration and the trustworthiness of product reviewers.

2.8 Foundations in Thai Text Mining

Srikamdee proposed a beginner-friendly framework for Thai text analysis, emphasizing the text cleaning process. This framework covers various aspects of text cleaning and presents results clearly. It includes a Thai sentiment analysis based on NLP combined with machine learning algorithms, designed to process Thai sentences across multiple domains without needing a dictionary or lexicon. The study tested ten well-known classifiers using TF-IDF in the feature extraction process. The datasets (Wisesight and 40 Thai Children's Tales) demonstrate the framework's applicability across multiple domains.

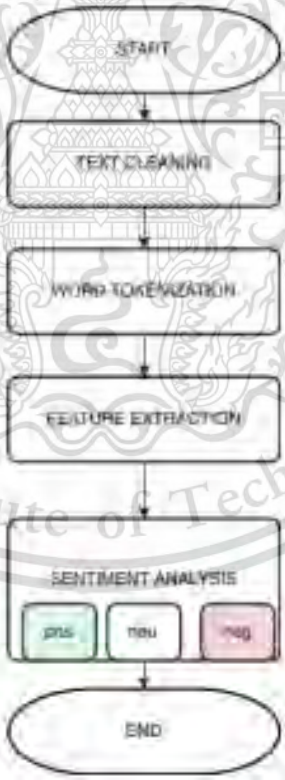

Figure 2.8 The proposed sentiment analysis method

Original data	Remove Special Character
ทำไมรถ isuzu มันไม่มีความนุ่นนวลกับการส่งกำลังเลย ครับ #Isuzu_D-Max	ทำไมรถisuzuมันไม่มีความนุ่นนวลกับการส่งกำลังเลย ครับIsuzuDMax
ไม่รู้จะแดกไรละเบื่อออ (@ เอ็มเค in Mueang Chiang Rai, Chiang Rai)	ไม่รู้จะแดกไรละเบื่อออเอ็มเค inMueangChiangRaiChiangRai
. เรื่อง ของ ผล ประโยชน์..ขัดกัน สิงห์..ช้าง..เบียร์ ลด ดราม่า...ชะ .	เรื่องของผลประโยชน์ขัดกันสิงห์ช้างเบียร์ลดดราม่าชะ

Figure 2.9 the example dataset after eliminating special characters

Original data	Text Normalization
ใช่ปะะะะะnarswantedอีกนะแล้วจะออกมาทำไมเวอร์ชั่น ไม่เข้าจายยยยยย	ใช่ปะnarswantedอีกนะแล้วจะออกมาทำไมเวอร์ชั่น ไม่เข้าจายยยยยย
เมื่อกี้ไปกินแล้วไม่ยักกะบอกว่าเป๊ปซี่ลดราคาพึ่งทำบัตร สมาชิกด้วยซ้ำ	เมื่อกี้ไปกินแล้วไม่ยักกะบอกว่าเป๊ปซี่ลดราคาพึ่งทำ บัตรสมาชิกด้วยซ้ำ
นี่ก็ทำนะน้ำอ้วนแต่เอาใส่เบียร์ช้างน้ำลิ้นจี่ดอยคำ เมาาาาาาาาาา	นี่ก็ทำนะน้ำอ้วนแต่เอาใส่เบียร์ช้างน้ำลิ้นจี่ดอยคำเมา

Figure 2.10 the example dataset after applying text normalization

Original data	Remove emoji icon and English word
ไม่รู้จะแดกไรละเบื่อออเอ็มเค inMueangChiangRaiChiangRai	ไม่รู้จะแดกไรละเบื่อออเอ็มเค
เสียใจจังเลยค่ะแต่ถ้าเปลี่ยนใจเมื่อไหร่อย่าลืมอินนีสฟรีนะคะ 😊🙏	เสียใจจังเลยค่ะแต่ถ้าเปลี่ยนใจเมื่อไหร่อย่าลืม อินนีสฟรีนะคะ
มอบความสุขผ่านเสียงเพลง 🎤🎧🎼 ลานเบียร์สิงห์ NANboxmusicnanfc	มอบความสุขผ่านเสียงเพลงลานเบียร์สิง

Figure 2.11 the outcome of removing non-Thai text from the sample dataset

Original data	Remove duplicate alphabet
เห้ยแกร**ssssss**นังอินนีสฟรีเทอว์จัมทำแบบนี้มะด้าย**ยยยยยยยย**	เห้ยแกรนังอินนีสฟรีเทอว์จัมทำแบบนี้มะด้าย
ว่าเป็นเมนส์ก็หงุดหงิดแล้วนะเจอกาวผ้าอนามัยแล้วแบบว้อย**ยยยยยยยยยยยนยยยยยยนยยน**	ว่าเป็นเมนส์ก็หงุดหงิดแล้วนะเจอกาวผ้าอนามัยแล้วแบบว้อยนยนยน
หมูสาขาเดอะมอบางแคเซนทรัลบางนาเล็กมาก**nnnn** อ้าวเฮ้ยไม่เหมือนที่คุยกันไว้นี่หว่า	หมูสาขาเดอะมอบางแคเซนทรัลบางนาเล็กมากอ้าวเฮ้ยไม่เหมือนที่คุยกันไว้นี่หว่า

Figure 2.12 The outcome of applying their approach on duplicate characters

Thai sentiment analysis was proposed in this study based on NLP combined with a machine learning algorithm. Their framework was designed to process Thai sentences in multiple domains without a dictionary or lexicon. This study tested the classification performance of ten well-known classifiers using TF-IDF in the feature extraction process. The results from testing on two domain datasets (Wisesight and 40 Thai Children's Tales) show that Logistic Regression and SVM yield an accuracy of 72% for the Wisesight dataset. For 40 Thai Children's Tales, Logistic Regression and SVM are the algorithms that provided the highest accuracy, up to 73%. Therefore, the proposed framework can be applied to multiple domains.

Chapter 3

Text Analysis and Visualization Framework

In this section, we propose a framework that centered around enhancing visualization techniques through the utilization of data obtained from governmental websites. Primarily, the focus lies on extracting relevant data from spreadsheets available on these websites, which contain valuable information regarding governmental activities and expenditures.

The initial phase involves extracting pertinent data from these spreadsheets, which are often abundant with information but may lack structure or organization conducive to meaningful analysis and also pdf file that contain text information. To address this, a series of preprocessing steps are undertaken. These include employing text processing techniques such as regular expressions (regex) and PyThaiNLP, which enable the cleaning, tokenization, and normalization of the textual data. Furthermore, the contextual understanding of sentences plays a crucial role in this process. By considering the context within which individual words or phrases are used, the research aims to extract key data points with higher accuracy and relevance. This contextual understanding is particularly important in the domain of governmental activities, where nuances in language can significantly impact the interpretation of data. The core objective of the methodology is to develop a framework that not only enhances the visualization of extracted data but also improves the extraction process itself. The original government visualization is provided as follow in fig.3.1 and fig.3.2

By integrating advanced text processing techniques with visualization tools, the proposed framework aims to provide better insights into governmental activities and expenditures. This framework will enable stakeholders to extract key data points more efficiently and visualize them in a manner that facilitates deeper understanding and analysis.

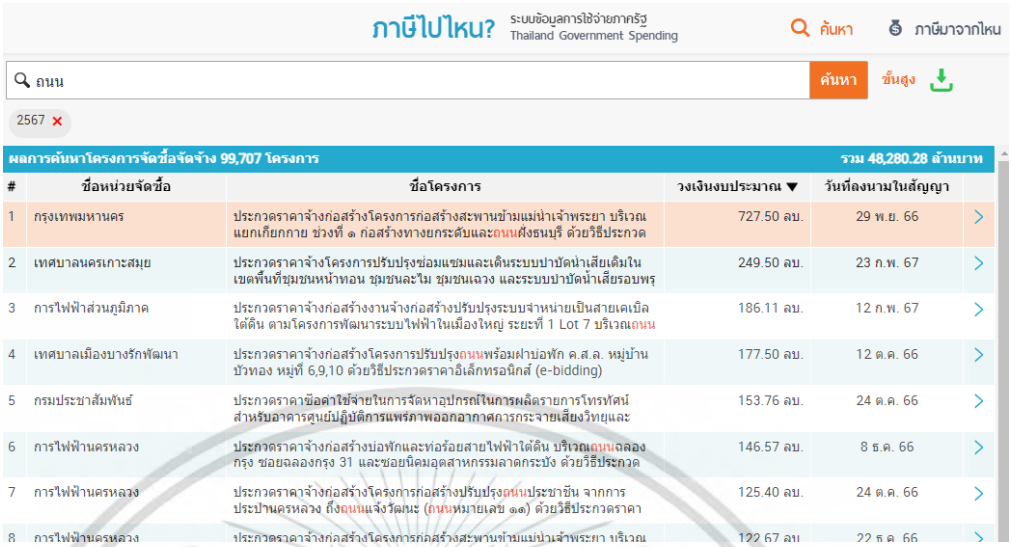

Figure 3.1 Original Government Website (1)

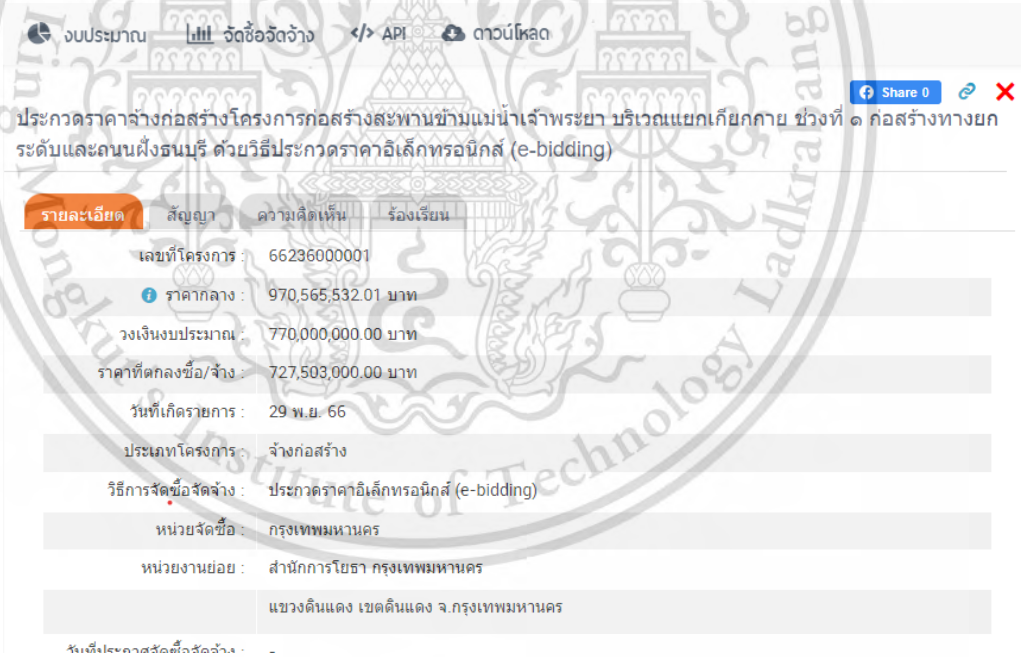

Figure 3.2 Original Government Website (2)

3.1 Keyword extraction

To extract monetary values from the textual data, a multi-step approach is employed.

3.1.1 Texts preprocessing to clean and structure the data.

This involves removing irrelevant information and standardizing the format of the text to facilitate further analysis shown in fig 3.3

ประกวดราคาจ้างก่อสร้างโครงการก่อสร้างสะพานข้ามแม่น้ำเจ้าพระยา บริเวณแยกเกียกกาย ช่วงที่ ๑ ก่อสร้างทางยกระดับและถนนผึ้งธนบุรี ด้วยวิธีประกวดราคาอิเล็กทรอนิกส์ (e-bidding)

Figure 3.3 Example of key information

3.1.2 Identify patterns indicating the presence of monetary values.

This includes identifying numerical sequences that are likely to represent monetary amounts. However, it's essential to recognize that numbers alone may not provide sufficient context as shown in fig 3.4. Therefore, additional steps are taken to ensure that the extracted monetary values are meaningful and interpretable. The data is extracted from spreadsheets using the xlsx Python library. This step involves accessing and importing the relevant data from the spreadsheet files containing information about road construction projects and their associated budgets.

'โครงการดิจิทัลวอลเล็ต' ที่ถูกใส่เข้าในงบประมาณรายการงบกลางแล้ว จำนวน 152,700 ล้านบาท

Figure 3.4 Information surrounding the monetary values is extracted to provide clarity and relevance

Contextual information surrounding the monetary values is extracted to provide clarity and relevance. This includes identifying the road construction

projects associated with the monetary values and the specific budget allocations for each project.

For example, if the extracted text mentions "Road 254 budget: 5 million baht", the methodology ensures that both the road name ("Road 254") and the monetary value ("5 million baht") are captured together, providing the necessary context for analysis.

3.1.3 Apply regular expressions (regex)

To identify specific patterns within the text. For instance, regex patterns are used to detect project names and currency values located at the end of numerical sequences. This helps in accurately extracting relevant information such as project names and associated budget amounts.

```
name_of_road = 'ถนนสุขุมวิท'
with open('datatest.csv', newline='', encoding='utf-8') as file:
    reader = csv.reader(file)
    next(reader)  # skip header row
    rows_containing_name_of_road = []
    for row in reader:
        match = re.search(rf'{name_of_road}\S*', row[1])
```

Figure 3.5 Using keyword with regex

3.1.4 Apply PyThaiNLP functions

For further analysis of the preprocessed text. Functions such as word frequency analysis, text summarization, and similarity calculations are applied to gain insights into the textual data. Word frequency analysis helps in identifying commonly occurring terms, while text summarization aids in condensing large volumes of text into concise summaries. Similarity calculations enable comparisons between different textual elements, facilitating the identification of similarities or differences between project descriptions.

```
rows_containing_name_of_road = []
for row in reader:
    match = re.search(rf'{name_of_road}\S*', row[1])
    if match:
        project_name = row[1]
        rows_containing_name_of_road.append(match.group())
#print(rows_containing_name_of_road)
rank_road = rank(rows_containing_name_of_road)
#print(rank_road)
```

Figure 3.6 Using rank function

3.1.5 Visualization

The final product of this study will be a web application designed to provide users with an intuitive interface for exploring and visualizing data related to road construction projects. The web application serves as a powerful tool for users to interactively explore and visualize data related to road construction projects, enabling them to gain valuable insights and make informed decisions based on the information provided by using chart.js from JavaScript.

In addition to the aforementioned functionalities, the web application incorporates the PyThaiNLP function called word similarity to enhance the user experience. This feature leverages word similarity algorithms to identify keywords or phrases that are semantically similar to the user's input.

3.3 Creating tag feature

3.2.1 Word Similarity Analysis

Upon receiving the user's input, the application utilizes the word similarity function from PyThaiNLP to analyze the semantic similarity between the input keyword or phrase and other words or phrases in the dataset. This

analysis identifies related terms or concepts that may be relevant to the user's search query.

3.2.2 Tag Generation

Based on the results of the word similarity analysis, the application generates a set of tags or keywords that are semantically similar to the user's input. These tags serve as additional search suggestions or refinements, helping users discover related information and insights.

```
if search_text is not None:
    list_positive = [search_text] if search_text is not None else []
    list_negative = []
    ans = wv.most_similar_cosmul(list_positive, list_negative)
```

Figure 3.7 Using word vector

3.2.3 Tag Display

The generated tags are displayed alongside the search results or visualization output, providing users with options to further refine their search or explore related topics. Users can click on these tags to dynamically update the displayed data and focus on specific areas of interest.

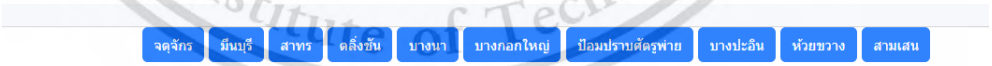

Figure 3.8 Display generated tag

By incorporating the word similarity analysis and tag generation feature, the web application enhances the user experience by facilitating more intuitive and efficient data exploration. Users are encouraged to discover new insights and uncover hidden patterns by exploring related keywords and concepts suggested by the application.

3.3 Data extraction from news websites into the web application

Additional steps and considerations are required due to the unstructured nature of the data. Here's how the process can be integrated into the application flow

3.3.1 Utilizing the BeautifulSoup library in Python

The web application scrapes data from news websites that contain topics related to government budgeting. This process involves accessing the HTML structure of the web pages and extracting relevant textual content. Since the data from news websites is unstructured, special attention is required to identify and extract pertinent information. This may involve parsing through various sections of the webpage, such as headlines, article bodies, and metadata, to capture relevant data related to government budgeting.

3.3.2 Adapting Regex Patterns

One of the challenges encountered when extracting data from news websites is the variability in text formatting and organization. To address this challenge, the regex patterns used for data extraction may need to be adjusted dynamically to accommodate variations in text structure and content.

By iteratively refining the regex patterns based on the observed data patterns from news websites, the web application ensures robust and accurate extraction of relevant information, despite the unstructured nature of the data.

Figure 3.9 Example of news website

3.4 Proposed Framework

This section is going to talk about steps of how to use the framework:

3.4.1 Data Integration

To ensure data integrity and easy accessibility, the extracted information is stored in CSV (Comma-Separated Values) files, a tabular structure, allowing easy import into any analytical tools including python editors, enabling easy integration of data into subsequent analysis processes in our framework and also, we implemented web scraping to efficiently extract data from the online sources as mentioned by using Python. Python is used due to the versatile and powerful programming language, facilitates the implementation of web scraping techniques, including the BeautifulSoup library, coupled with requests, as well as other robust combinations of data parsing techniques to process and extract data from HTML and XML files.

```
file_path_csv = os.path.join(os.path.dirname(os.path.abspath(__file__)), 'datatest.csv')
with open(file_path_csv, 'r', encoding='utf-8') as f:
```

Figure 3.10 csv file integration

3.4.2 Textual Data Identification

To effectively analyze government spending budgets, relevant data was identified and collected. You need to identify the topic that you are interested in since web scraping can be used as a valuable tool for this, especially for data that is dynamic and frequently updated, the data has been scraped from the government official websites, the Thailand Government Spending website (https://govspending.data.go.th/), which also provides an API, making it easier for data extraction.

3.4.3 Data Preprocessing and Keyword Recognition

Because understanding what is going on in the datasets is crucial, data has been studied to get an idea, which can lead to meaningful analysis. In the case of our government spending budgets, patterns and keywords that are deemed relevant were specified.

3.4.4 Keywords Preceding

Since in this study we are focusing on budget. The budget numbers: (or “จัดการ" in Thai), and “budgeted for” (or "งบประมาณสำหรับ" in Thai), or similar phrases. These words could be identified in Thai text.

```
keyword_work = 'จ้างซ่อมแซมถนนลูกรัง'
tt_work_money = 0
with open('datatest.csv', 'r', encoding='utf-8') as f:
    reader = csv.reader(f)
    next(reader) # skip the header row
    matching_rows_work = []
    # Check if the value in column 3 matches the keyword
    for row in reader:
        match = re.search(rf'{keyword_work}\S*', row[1])
```

Figure 3.11 Adapting keyword and regular expression

3.4.5 Contextual Analysis

We also consider surrounding texts, such as project descriptions, departmental mentions, and date/time/timestamps as well as other important

attributes, which potentially can provide additional context for a better understanding.

3.4.6 Visualization

The final step is to use web application tool to display in graph as dashboard from text-based data.

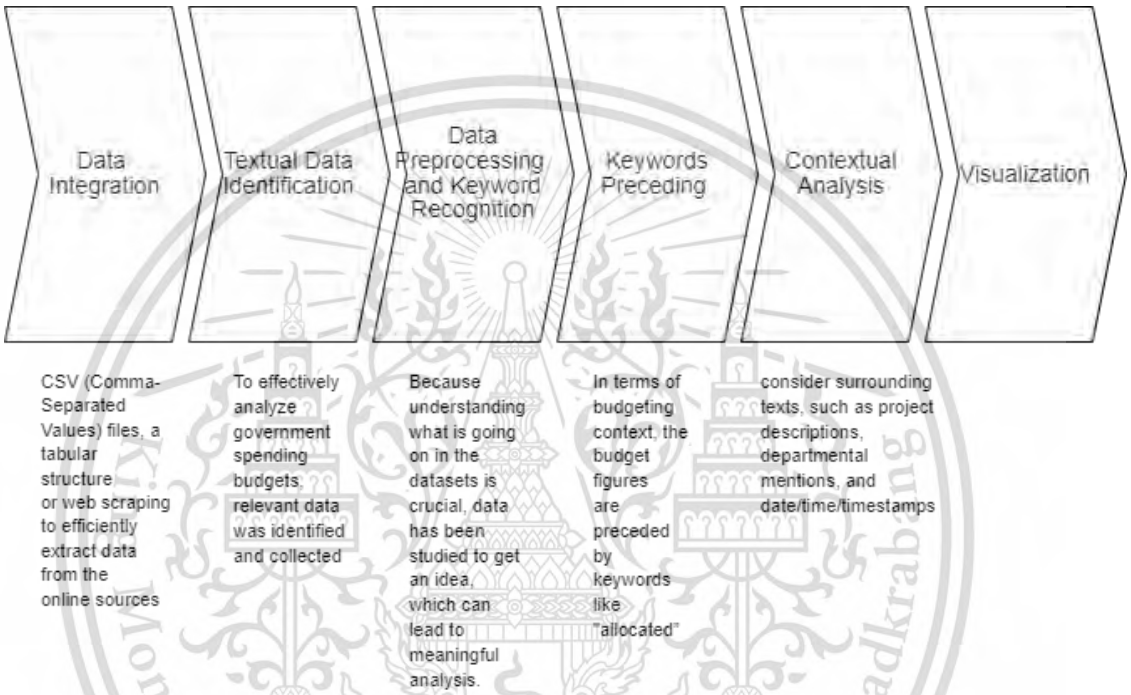

Figure 3.12 Proposed framework.

The algorithms 1 and 2 below describe how these components are put together into work, which uses governmental budget documents and budgets for a specific road as an example:

Algorithm 1: Searching and Analyzing Financial Data

Input: HTTP request with a search text parameter.

Output: Rendered HTML page with data visualizations.

1. Get Search Texts from the Request
2. Read Data from CSV, JSON files
3. Search and Collect Relevant Data
4. If a match is found:
5. Extract project details and financial information.
6. Update total work money (tt_work_money) and
7. project count (count_project).
8. Append project details to matching_rows_work.
9. Append date and money details to date_and_money.
10. Wordvector Similarity:
11. If search_text is not empty:
12. Use WordVector to find words similar to search_text.
13. Generate tags from the results.
14. Format Data for Charts:
15. Extract labels and values from the paginated data for
16. uses in charts.
17. Render HTML Page:
18. Pass the formatted data, pagination information, date
19. and money details, project count, and generated tags to
20. the template context.
21. Render the 'index.html' template with the context.

Algorithm 2: Ranking Projects Related to a Specific Road

Input:

name_of_road: The name of a road, e.g., 'ถนนสุขุมวิท','ถนนลาดกระบัง'

'datatest.csv': A CSV file containing project data.

Output:

rank_road: A ranking of projects related to the specified road.

Initialization:

1. Set name_of_road to 'ถนนสุขุมวิท'.

2. Open and read the 'datatest.csv' file, excluding the header row.
3. Initialize an empty list rows_containing_name_of_road to store rows related to the specified road.

Search for Relevant Projects:

4. For each row in the CSV file:
5. Check if name_of_road matches a portion of the
6. project name using regular expressions.
7. If a match is found:
8. Extract the project name.
9. Append the matched portion to
10. rows_containing_name_of_road.

Ranking Projects:

11. Utilize a function (rank()) to rank the projects in rows_containing_name_of_road.
12. Store the ranking result in rank_road.

Output: rank_road now contains the ranking of projects related to the specified road.

Chapter 4

Main results and discussion

In this section, the practical application of the proposed methodology is demonstrated by using a case study on estimating allocated budgets for road construction in Thailand. The objective was to showcase the effectiveness of our Thai analysis and visualization framework in processing and understanding of governmental budget allocation and expenditure data patterns.

4.1 Main results

The data analysis reveals the total amount of money allocated to projects containing specific keywords related to government budgeting. This information provides a comprehensive overview of budget allocations that may not be readily available from the original data sources. By analyzing the data, trends emerge regarding how government funds are invested across different locations and projects. This insight enables stakeholders to understand the prioritization of budget allocations and identify areas of focus within the government's budgetary initiatives.

While the Thai Governmental Spending website offers valuable data, the available visualization tools have limitations in terms of depth and user flexibility as shown in Figure 4.1 Many of the visualizations are text-based and fail to provide a comprehensive understanding of the intricacies of budget allocation and expenditure.

By applying our Thai text analysis and visualization framework developed in this research, we aim to address these limitations and enhance the user experience. We believed that transforming the raw unstructured textual data into interactive charts and graphs could lead to a more intuitive and insightful representation of governmental spending patterns. This, in turn, could allow users to make informed interpretations and decisions based on the data more effectively.

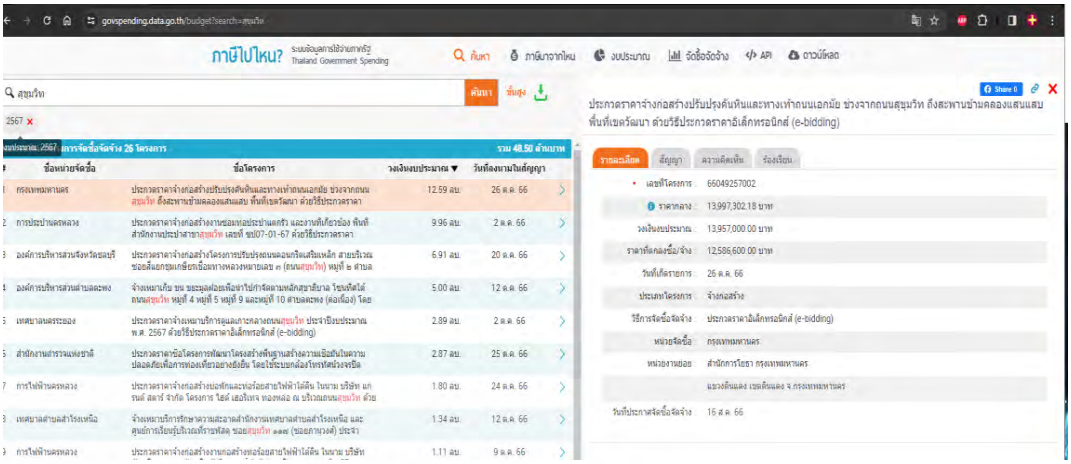

Figure 4.1 Sample visualization from the Thai Governmental Spending website

The visualization techniques were employed to convert the unstructured data into interactive graphical representations. Different types of visualizations were created, such as bar charts, line graphs, pie charts and donut charts, to highlight key aspects of governmental spending, including budget distribution across departments, expenditure trends over time, and project completion rates. Example screenshots which use estimating allocated budget for road construction are described in the following figures 4.2-4.9

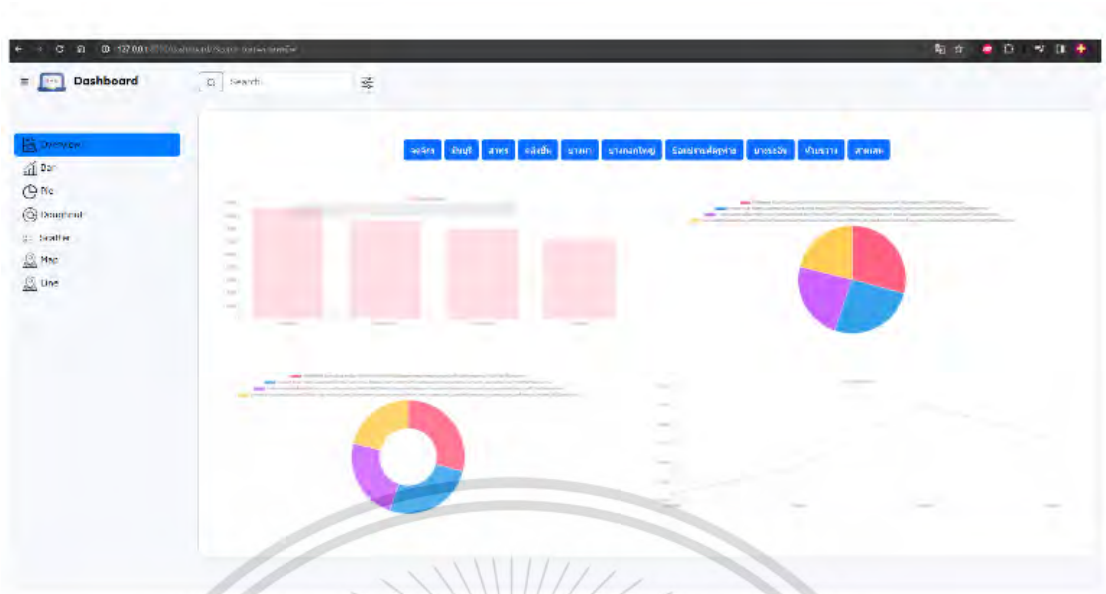

Figure 4.2 Main Dashboard

The main dashboard displaying all the visualization in one place so user can see all the detail and insight in high level perspective.

Figure 4.3 Search bar feature

On the top left there is a search bar where user can input any keyword, they want to look for visualization and insight. The keyword can be any name of roads or project names.

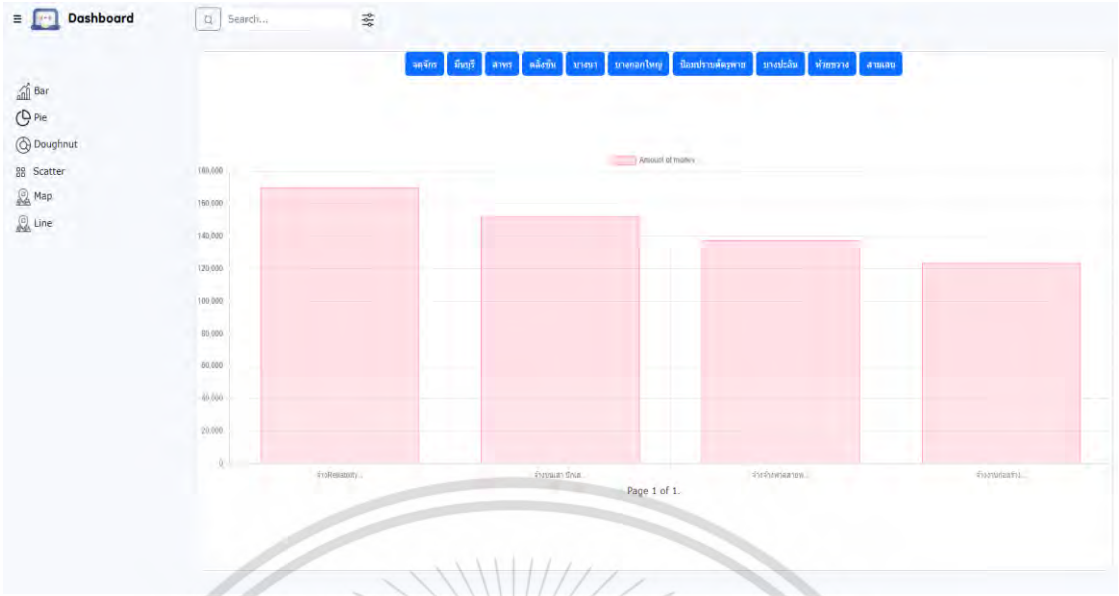

Figure 4.4 Sample generated bar chart

After the user input, the visualization will be generated base on the keyword input.

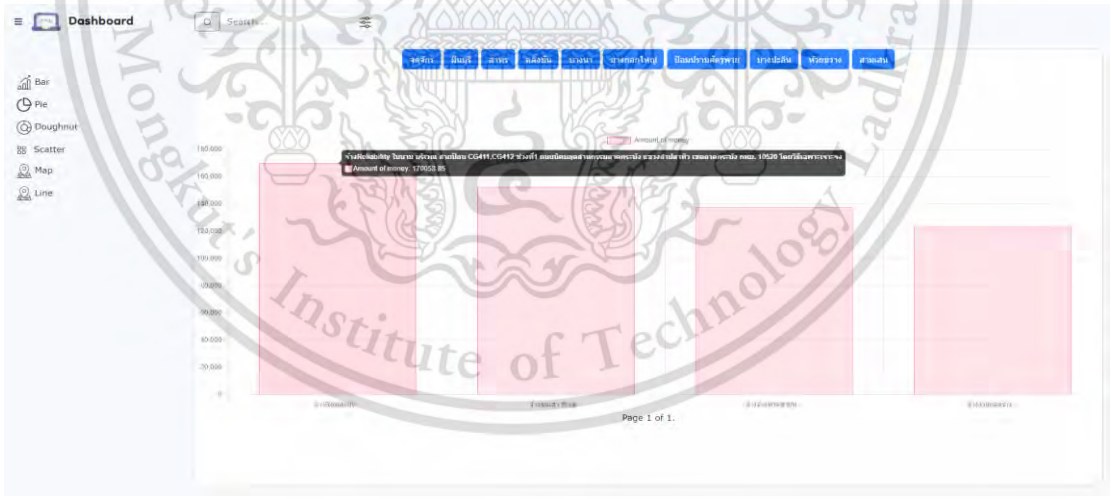

Figure 4.5 Sample generated bar chart with interactive details

For the bar chart user can interact with each chart showing more details such as name of project, the full name of project and roads, amount of money using in each project and each of them will be sorted.

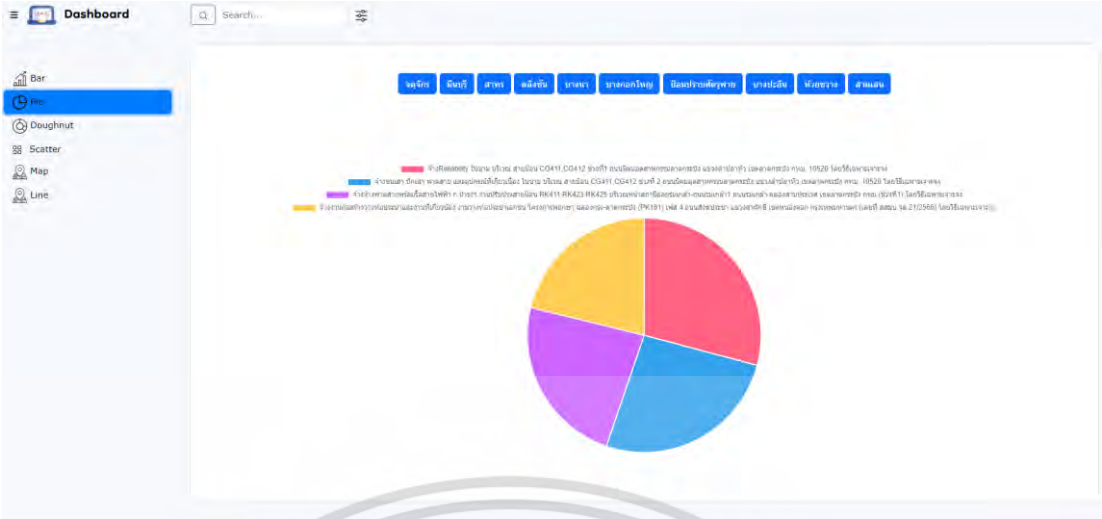

Figure 4.6 Sample generated pie chart with details

The pie chart visualization is better when look for the portion perspective.

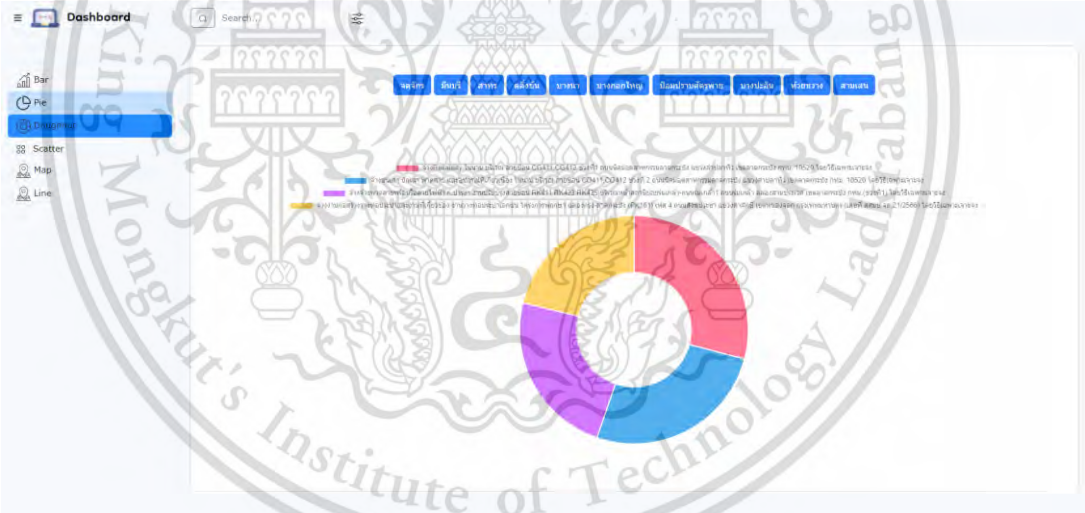

Figure 4.7 Sample generated doughnut chart with details

The doughnut chart visualization is better when look for each part perspective.

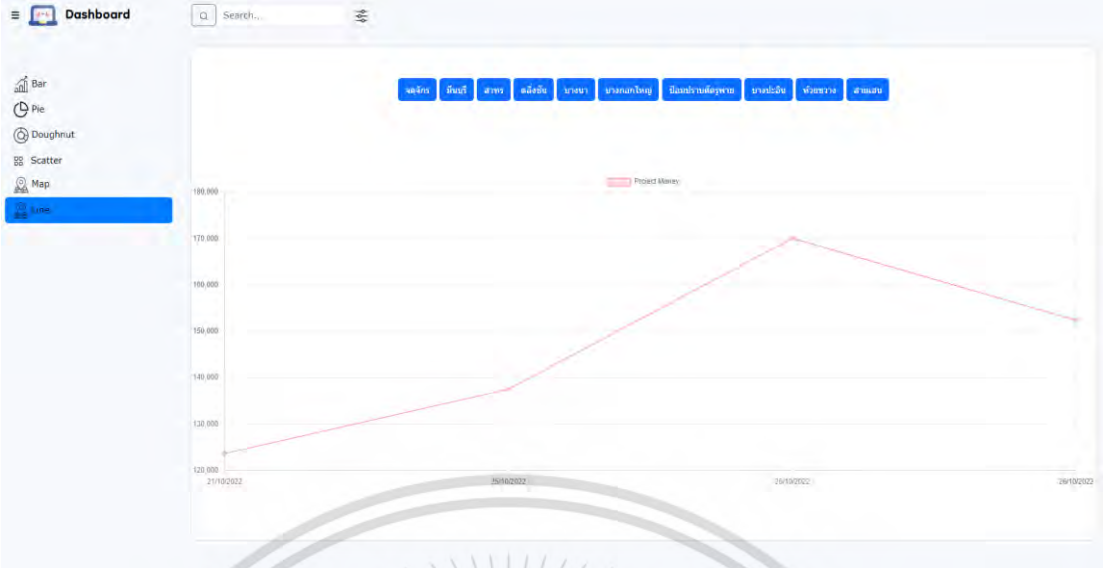

Figure 4.8 Sample generated line chart

Line chart for graphical representations that succinctly illustrate trends and patterns in numerical data

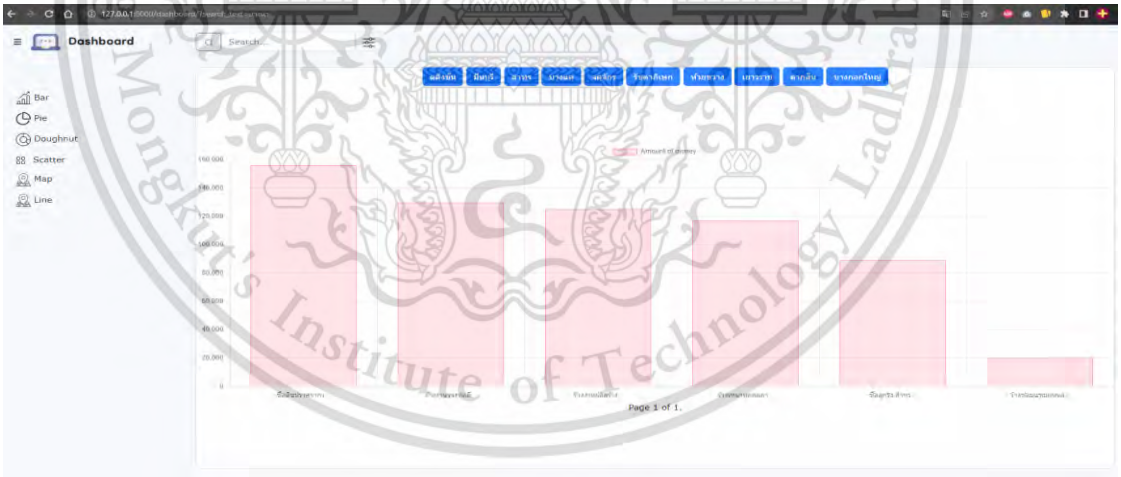

Figure 4.9 Sample bar chart with tags generated from program

On the top of the chart there will be generated tags for user to continue or go deeper for insight that similar to keyword input.

4.2 Discussion

Through the application of our Thai analysis and visualization framework, we summarize our outcomes as follows

1) Deeper Insights: Interactive charts allowed users to delve deeper into specific aspects of governmental spending. For instance, users could click on specific bars or segments to access detailed information about individual projects and associated expenditure.

2) User-Friendly Interface: The visualizations transformed complex numerical/unstructured data into visual forms that were easy to interpret, even for non-experts. This increased accessibility can encourage more users to engage with and understand governmental spending patterns.

3) Identifying Trends: Visualization of expenditure trends enables users to identify patterns and anomalies, aiding in the identification of areas that may require further investigation or adjustment in resource allocation.

To compare our results with other approaches, the Governmental website is compared as shown in Fig. 4.10 A table summarizing the key features between the two is described in Table 4.1

Table. 4.1 Comparing our approach with governmental website

Feature	Our approach	Governmental website
1. Flexibility	Using keyword search, more flexible.	Predefined keyword only, not flexible.
2. Interactivity with datasets	Allowing “drill down” feature so data can be explored more effectively.	One-level only, data cannot be drilled down further.
3. Extendable since program is implemented by python & open-source	Yes.	No.
4. Supporting variety of data (not just budgeting data)	Yes, new data can be imported/ingested to enhance analyses.	No, limited to budgeting data only.
5. Compatible with other data science tools in the future	Yes.	No.

Comparing our approach with governmental website

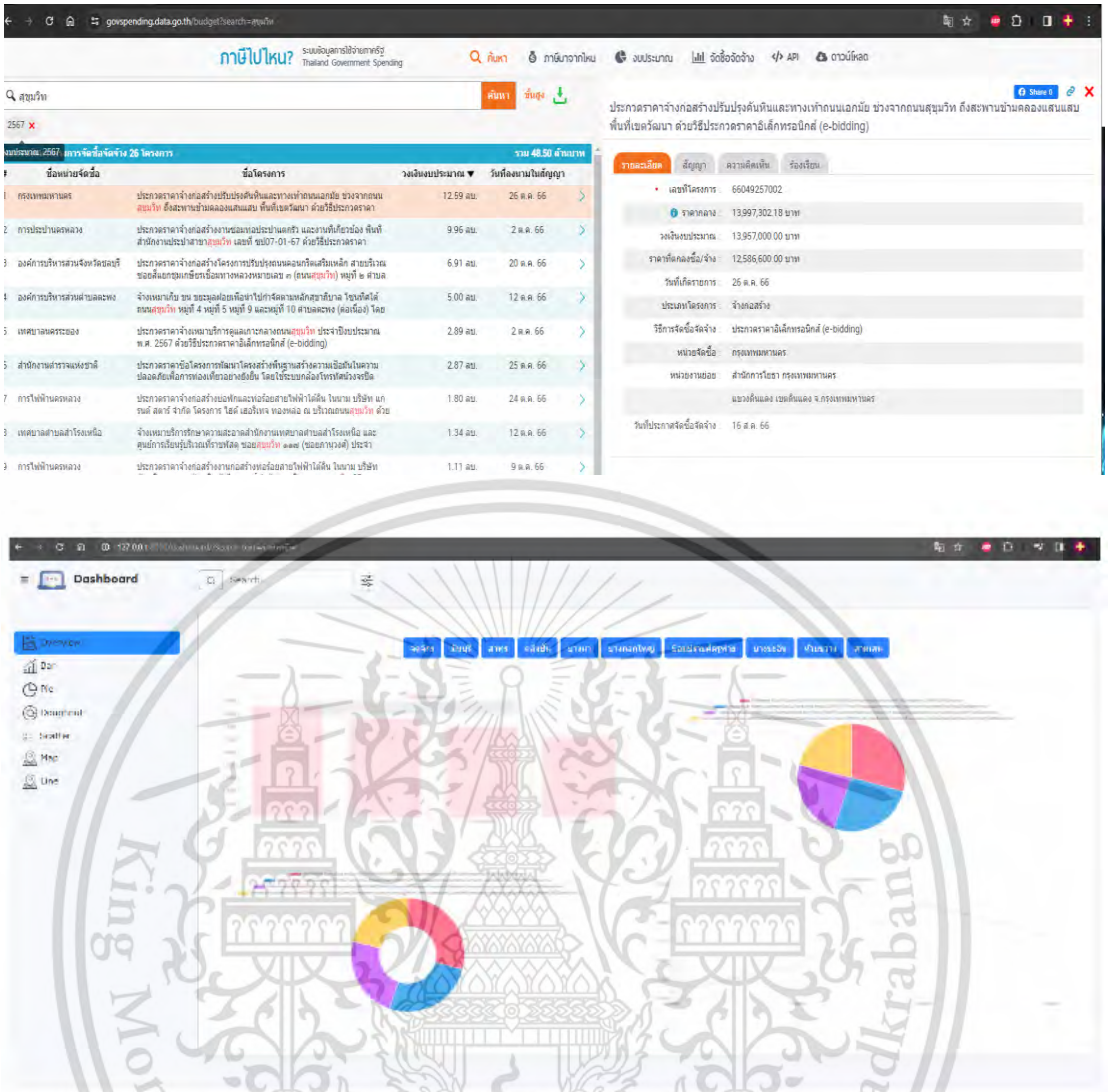

Figure 4.10 Comparing both visualizations between other method (top) and our method (bottom)

Chapter 5

Conclusions and suggestions

In this research, we tackled the complexities of processing Thai language texts, especially from social media and online platforms. This study introduced an analysis and visualization framework specifically designed for Thai text data, utilizing natural language processing (NLP) and visualization techniques to enhance understanding and insights from large datasets.

5.1 Conclusions

Throughout our research process, we encountered several challenges related to Thai vocabulary and sentence structures. The Thai language's unique grammatical and linguistic characteristics required careful consideration. The effectiveness of keyword recognition is closely linked to the precise definition of the topic. Careful selection of keywords is crucial during the initial phase of studying the chosen topic or theme, as it impacts the relevance and accuracy of the analysis and visualization.

Our proposed analysis and visualization framework facilitates the analysis and visualization of large unstructured Thai textual data. The findings suggest that our framework enhances Thai language processing and improves the understanding of text data in a timely manner. However, challenges such as addressing non-standard language variations and the ongoing evolution of the Thai language persist.

5.1.1 The main findings and conclusions from the research

1) Effective Framework for Thai Text Data Analysis: We successfully developed a robust framework for analyzing and visualizing Thai text data. This framework addresses the unique grammatical and lexical challenges inherent in the Thai language, including complex script representation and lexical ambiguity.
2) Enhanced Data Insight: Our framework improved the extraction and comprehension of information from Thai textual data. By using techniques such as tokenization, morphological analysis, part-of-speech

tagging, named entity recognition, and sentiment analysis, we were able to handle the intricacies of the Thai language more effectively.

3) Visualization of Budget Trends: The visualization techniques employed, including interactive graphs, word clouds, and topic modeling, provided intuitive and insightful representations of the processed data. These visualizations were instrumental in capturing crucial information from social media discussions and online content related to government budgeting.

4) Case Study on Road Construction Budgeting: The practical application of the proposed framework was demonstrated through a case study on estimating allocated budgets for road construction in Thailand. This case study highlighted the framework's effectiveness in processing and understanding governmental budget allocation and expenditure data patterns.

5.2 Suggestions

Based on the findings and challenges encountered during this study, several suggestions for future research and practical applications are proposed:

1) Continuous Data Updates
2) Broaden Domain Applications
3) User-Centric Enhancements
4) Collaboration and Knowledge Sharing
5) Comprehensive Tool Documentation

For future work, the visualizations generated by the system may not fully capture the dynamic nature of the underlying phenomena. Changes or updates in the dataset may not be reflected in real-time, and the visualizations may not adapt to new patterns or trends as they emerge. This static nature of the visualizations limits their utility for ongoing monitoring or decision-making processes that require up-to-date information. To mitigate this limitation, it is essential to periodically update the

dataset used for analysis and visualization. This may involve integrating mechanisms for automatically fetching and processing new data from governmental and news sources. Additionally, implementing feedback loops or mechanisms for user input can help improve the relevance and accuracy of the visualizations over time.

In conclusion, this study provides a solid foundation for Thai text data analysis and visualization, offering valuable insights and practical tools for various applications. By addressing the identified challenges and embracing the suggested improvements, future research can further enhance the capabilities and impact of Thai text processing frameworks.

References

M. Masdisornchote. "A sentiment analysis framework in implicit opinions for Thai language." In IECON 2015-41st Annual Conference of the IEEE Industrial Electronics Society, pp. 000357-000361. IEEE, 2015.

A. Willem, G. Sukhwir, D. Suhartono, and L. Christina. "Detecting Racist and Bad Words Using Text Mining in Social Media." In 2021 8th International Conference on Information Technology, Computer and Electrical Engineering (ICITACEE), pp. 85-88. IEEE, 2021.

K. Katchapakirin, K. Wongpatikaseree, P. Yomaboot, and Y. Kaewpitakkun. "Facebook social media for depression detection in the Thai community." In 2018 15th international joint conference on computer science and software engineering (jcsse), pp. 1-6. IEEE, 2018.

P. Zhang, and J. Wang. "Optimization research of short text semantic clustering based on the social media." In 2017 IEEE 3rd Information Technology and Mechatronics Engineering Conference (ITOEC), pp. 717-721. IEEE, 2017.

S. Srikamdee, U. Suksawatchon, and J. Suksawatchon. "Thai sentiment analysis for social media monitoring using machine learning approach." In 2022 37th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC), pp. 1-4. IEEE, 2022.

A. Kongthon, A Framework for Managing R&D for Thai Research Community Using Text Information Exploitation, PICMET 2008 Proceedings, 27-31 July, Cape Town, South Africa (c) 2008 PICMET

A. Dewinta, and M. I. Irawan. "Customer complaints clusterization of government drinking water company on social media twitter using text mining." In 2021 3rd East

Indonesia Conference on Computer and Information Technology (EIConCIT), pp. 338-342. IEEE, 2021.

S. Jeelall, and S. Cheerkoot-Jalim. "HealthMine: A Tool for Social Media Text Mining in Health." In 2020 3rd International Conference on Emerging Trends in Electrical, Electronic and Communications Engineering (ELECOM), pp. 53-57. IEEE, 2020.

Box Blog, “BoxWorks 2023: Unlock the value of your content”, retrieved Nov. 20, 2023 from: https://blog.box.com/

IDC White Paper, sponsored by Box, “Untapped Value: What Every Executive Needs to Know About Unstructured Data,” Doc #US51128223, retrieved August 2023 from: https://www.box.com/resources/unstructured-data-paper

Author biography

Name	Mr.Chanwit Kongthong		
Date of Birth	13 March 1998		
Address	18/121 Bangna Bangkok 10260		
Education	(2014) Bachelor of Science in Computer Science	GPA	2.84
	King Mongkut's Institute of Technology Ladkrabang		
	(2018) Master of Science in Computer Science	GPA	3.50
	King Mongkut's Institute of Technology Ladkrabang