**Document Classification in Graph Theory**



Session: 2021 – 2025

**Submitted by:**

**Huzaifa Majeed 2021-CS-72**

**Rana Hamid Shahid 2021-CS-84**

**Submitted to:**

**Waqas Ali**

Department of Computer Science

**University of Engineering and Technology**

Contents

[**Introduction**](#_llq730q6pbcd) **2**

[**Data Collection and Preparation**](#_6tn6n1m4s75j) **3**

[**Graph Construction**](#_e9hhdqwqcttf) **3**

[**Feature Extraction via Common Subgraphs**](#_vh6lm3wfsyu3) **3**

[**Classification with KNN**](#_6zn0qa7r4wtt) **3**

[**Methodology Details**](#_7fqn8x91gyhq) **3**

[**Evaluation**](#_i7o4vdgwfh0) **4**

[**Challenges and Improvements**](#_le7spxamjk5w) **4**

[**Implications and Future Work**](#_hell6qom9vv0) **4**

# 

# 

# 

# 

# 

# 

# Introduction

Document classification is a vital task in natural language processing (NLP), involving the categorization of text documents into predefined topics. Traditional methods often use vector-based representations, which may overlook complex relationships within text data.

In this project, we explore a graph-based approach for document classification using the K-Nearest Neighbors (KNN) algorithm with a distance measure based on maximal common subgraphs (MCS). By representing documents as graphs and extracting features from common subgraphs, we aim to capture inherent content relationships for more accurate classification.

The report details our methodology, including data collection, graph construction, feature extraction using common subgraph mining, KNN implementation, and performance evaluation metrics. We also discuss challenges, improvements, and implications for advancing document classification in NLP.

# Data Collection and Preparation

For this project, we collected text data from three assigned topics, each containing 15 pages with approximately 300 words per page. We divided the dataset into a training set (12 pages per topic) and a test set (3 pages per topic) to facilitate model training and evaluation.

# Graph Construction

Each page of text was represented as a directed graph using the networkx library in Python. The nodes of the graph corresponded to unique terms (words) extracted from the text after preprocessing steps such as tokenization, stop-word removal, and stemming. Edges between nodes were created to denote the sequence of terms in the text.

# Feature Extraction via Common Subgraphs

To extract features for document classification, we utilized frequent subgraph mining techniques on the training set graphs. Common subgraphs were identified across graphs related to the same topic. These common subgraphs served as features capturing shared content and structure within the documents.

# Classification with KNN

We implemented the K-Nearest Neighbors (KNN) algorithm for document classification, using a distance measure based on the maximal common subgraph (MCS) between document graphs. The MCS distance calculated the similarity between graphs by evaluating their shared structure. Test documents were classified based on the majority class of their k-nearest neighbors in the feature space defined by the common subgraphs.

# Methodology Details

* Graph Construction: Each document was processed to extract terms (words), which were then used as nodes in a directed graph. Edges were added between nodes to represent the sequential relationships between terms in the text.
* Common Subgraph Identification: Frequent subgraph mining techniques were applied to the training set graphs to identify frequently occurring subgraphs. These subgraphs represented common patterns shared among documents within the same topic.
* KNN Implementation: The MCS distance was computed between each test document and the training set documents. The KNN algorithm then identified the k-nearest neighbors based on the MCS distance and assigned the test document to the majority class among these neighbors.

# Evaluation

We evaluated the performance of our graph-based classification system using standard metrics such as accuracy, precision, recall, and F1-score. Additionally, we plotted a confusion matrix to visualize the classification results across different topic classes.

# Challenges and Improvements

Challenges encountered during the project included:

* Graph Representation Complexity: Managing large graphs efficiently, especially when dealing with multiple documents and frequent subgraph mining.
* Parameter Tuning: Optimizing parameters such as the number of neighbors (k) in KNN and the threshold for subgraph frequency.

Potential improvements to the approach:

* Enhanced Subgraph Mining: Explore advanced subgraph mining techniques to capture more complex patterns in document graphs.
* Graph Embeddings: Use graph embeddings to represent document graphs in a continuous vector space for improved classification.

# Implications and Future Work

The findings of this project highlight the effectiveness of using graph-based representations and MCS distance for document classification. Graph-based methods offer a unique perspective for capturing document structure and content relationships, potentially outperforming traditional vector-based approaches.

Future work could focus on scaling the approach to larger datasets, integrating semantic information into graph representations, and exploring ensemble methods combining graph-based and vector-based classifiers for robust document classification systems.

In conclusion, leveraging graph-based representations and common subgraph features presents promising avenues for advancing document classification techniques, offering insights into document content structure that traditional methods may overlook.