**ABSTRACT**

**Digital data in the form of text documents is rapidly growing. Analysing such data manually is a tedious task. Data mining techniques have been around to analyse such data and bring about interesting patterns. Many existing methods are based on term-based approaches that can’t deal with synonymy and polysemy. Moreover they lack the ability in using and updating the discovered patterns. Zhong et al. proposed an effective pattern discovery technique. It discovers patterns and then computes specificities of patterns for evaluating term weights as per their distribution in the discovered patterns. It also takes care of updating patterns that exhibit ambiguity which is a feature known as pattern evolution. In this paper we implemented that technique and also built a prototype application to test the efficiency of the technique. The empirical results revealed that the solution is very useful in text mining domain.**

**Text mining, also referred to as text data mining, roughly equivalent to text analytics, refers to the process of deriving high-quality information from text. High-quality information is typically derived through the devising of patterns and trends through means such as statistical pattern learning. Many data mining techniques have been proposed for mining useful patterns in text documents. However, how to effectively use and update discovered patterns is still an open research issue, especially in the domain of text mining. Since most existing text mining methods adopted term-based approaches, they all suffer from the problems of polysemy and synonymy.**

**Over the years, people have often held the hypothesis that pattern (or phrase)-based approaches should perform better than the term-based ones, but many experiments do not support this hypothesis. This paper presents an innovative and effective pattern discovery technique which includes the processes of pattern deploying and pattern evolving, to improve the effectiveness of using and updating discovered patterns for finding relevant and interesting information. Substantial experiments on RCV1 data collection and TREC topics demonstrate that the proposed solution achieves encouraging performance.**

**INDEX**

**Chapter No. Particulars Page No.**

**Abstract i List of Figures iv List of Tables vi**

1. **Introduction 1**
   1. Related Work 1
   2. Pattern Taxonomy Model 2
2. **Software Requirements Specification 3**
   1. Introduction 3
   2. Overall Description 4
   3. System features 6
   4. Other Non-Functional Requirements 7
3. **Analysis 9**
   1. Existing System 9
   2. Drawbacks of Existing System 9
   3. Proposed System 9
   4. Merits of Proposed System 10
   5. Feasibility Study 10
   6. Scope of the Project 11
4. **Requirements 12**
   1. Software Requirements 12
      1. Features of .NET Framework 12
      2. The .NET Framework 12
      3. The .NET Class Library 13
   2. Hardware Requirements 14

**Chapter No. Particulars Page No.**

1. **Design 15**
   1. Unified Modeling Language Diagrams 15

5.1.1 Class Diagram 16

5.1.2 Use Case Diagram 17

5.1.3 Activity Diagram 19

5.1.4 Sequence Diagram 20

5.1.5 Collaboration Diagram 22

5.2. Database Tables 23

**6** **Implementation 24**

6.1 Modules and description 24

6.2 Technologies used 26

6.2.1 ASP.NET 26

6.2.2 ASP.NET Web Sites 28

6.2.3 ASP.NET API reference 29

6.2.4 ASP.NET Life Cycle 31

6.2.5 C# 33

6.2.6 SQL Server 37

6.3 Screenshots 39

**7** **Testing 51**

7.1. Purpose 51

7.2. Dimensions of Testing 51

7.3. Types of Testing 51

7.4. Test Cases 54

**8**  **Conclusion and Future Enhancements 56**

**REFERENCES 57**

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Fig .No** | **Name Of Figure** | **PAGE NO** |
| 3.1 | Document Indexing | 10 |
| 5.1 | Class Diagram | 16 |
| 5.2 | Use case Diagram | 18 |
| 5.3 | Activity Diagram | 19 |
| 5.4 | Sequence Diagram | 21 |
| 5.5 | Collaboration Diagram | 22 |
| 6.1 | System Architecture | 26 |
| 6.2 | ASP.NET Page Life Cycle | 33 |
| 6.3 | .Net Framework | 35 |
| 6.4 | Home page | 39 |
| 6.5 | Registration page | 40 |
| 6.6 | Error messages on not entering registration details | 41 |
| 6.7 | Message displayed when registration is successful | 41 |
| 6.8 | Error message on not filling details for login | 42 |
| 6.9 | User Account | 43 |
| 6.10 | User can download full files by selecting file name | 44 |
| 6.11 | User can search for files by entering terms or file-text | 45 |
| 6.12 | User can search a file and download it to the system | 46 |
| 6.13 | Error message on entering invalid admin details | 47 |
| 6.14 | Admin can view all details of the registered users | 47 |
| 6.15 | Viewing list of all uploaded files | 48 |
| 6.16 | Message when file is successfully uploaded by user | 49 |
| **Fig No.** | **Name Of Figure** | **Page No.** |
| 6.17 | Error message when text field left empty while searching a file | 49 |
| 6.18 | Files stored in encrypted form in database | 50 |
| 6.19 | User registration details in database | 50 |

**LIST OF TABLES**

**Table no. Particulars Page no.**

5.1 User\_Table 23

5.2 File\_Table 23

7.1 Test cases 55