Multi-Criteria Mapping Based on Support Vector Machine and Cluster Distance

Master Thesis

for

the fulfillment of the academics degree

M.Sc. in Automotive Software Engineering

Faculty of Computer Science
Professorship of Computer Engineering
August 2016

Submitted by: Vishwa Shanthi Eerla, Mat.Nr:356327
Supervisor: Prof. Dr. Wolfram Hardt
            Dipl.-Inf. Daniel Reißner
Vishwa Shanthi Eerla (vishwa-shanthi.eerla@s2013.tu-chemnitz.de)

Multi-Criteria Mapping Based on SVM and Cluster Distance

Master Thesis, Technische Universität Chemnitz, August 2016
Acknowledgement

At this point, I would like to thank all the people who helped me to complete this thesis. First, I want to appreciate Prof. Dr. Wolfram Hardt, Faculty of Computer Science, Technische Universität Chemnitz, for giving me the opportunity to work under his supervision in this university with the title of “Multi-Criteria Mapping Based on Support Vector Machine and Cluster Distance.” The topic was very interesting, and I enjoyed on working it a lot. Here, I would like to take this opportunity to thank especially my thesis supervisor Dipl.-Inf. Daniel Reißner, Researcher in Technische Universität Chemnitz for his invaluable guidance, useful ideas, pragmatic suggestions, and precise contribution. This research work wouldn't have completed on time without his guidance.

I would like to thank my friends for encouragement and reviewing my document. The last but not the least I would like to thank all my friends and family for asking me for the progress of this work and thus providing a kind of staying power.
# Table of Contents

1. **Introduction** ...................................................................................................................... 1
   1.1. Motivation ......................................................................................................................... 1
   1.2. Problem Description ......................................................................................................... 2
   1.3. Project Information ......................................................................................................... 2
   1.4. Outlines of Thesis ........................................................................................................... 2
   1.5. Document Layout ........................................................................................................... 3

2. **Basic Information** ............................................................................................................. 4
   2.1. OCR (Optical Character Recognition) ............................................................................... 4
       2.1.1. Construction of OCR System .................................................................................. 5
   2.2. Tesseract .......................................................................................................................... 10
       2.2.1. Architecture ............................................................................................................... 10
       2.2.2. Line and Word Finding ........................................................................................... 11
       2.2.3. Word Recognition .................................................................................................... 13
       2.2.4. Static Character Classifier ......................................................................................... 14
       2.2.5. Adaptive Classifier .................................................................................................. 16
   2.3. Open CV ........................................................................................................................... 17
   2.4. Support Vector Machines .............................................................................................. 18
   2.5. Cluster .............................................................................................................................. 19

3. **State of the Art** .................................................................................................................. 24
   3.1. Text Mining ..................................................................................................................... 24
       3.1.1. Text Retrieval Methods ............................................................................................ 25
       3.1.2. Cluster Analysis ....................................................................................................... 25
   3.2. Machine learning ............................................................................................................. 30
       3.2.1. Multilayer Perceptron (MLP) .................................................................................. 31
       3.2.2. Radial Basic Function (RBF) .................................................................................. 33
   3.3. Support Vector Machine Module ..................................................................................... 34
       3.3.1. SVM in Text Detection and Character Extraction .................................................... 36
   3.4. Clustering ......................................................................................................................... 40

4. **Concept** ............................................................................................................................... 43
   4.1. OCR Detection ............................................................................................................... 43
   4.2. SVM Module ................................................................................................................... 45
   4.3. Cluster Module ............................................................................................................... 47
   4.4. Connectivity Check ......................................................................................................... 47
   4.5. Split rules ......................................................................................................................... 48
   4.6. Evaluation Module ......................................................................................................... 49
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.7</td>
<td>Organization of workflow first page</td>
<td>49</td>
</tr>
<tr>
<td>4.8</td>
<td>Organization of workflow of performance sheet</td>
<td>50</td>
</tr>
<tr>
<td>4.9</td>
<td>Online step organization</td>
<td>50</td>
</tr>
<tr>
<td>4.10</td>
<td>Hardware Module</td>
<td>51</td>
</tr>
<tr>
<td>4.11</td>
<td>Multi-Criteria Mapping Based on SVM and Clustering Methods</td>
<td>51</td>
</tr>
<tr>
<td>5.</td>
<td><strong>Implementation and Realization</strong></td>
<td>55</td>
</tr>
<tr>
<td>5.1</td>
<td>SVM Formulation</td>
<td>57</td>
</tr>
<tr>
<td>5.2</td>
<td>Cluster Connectivity</td>
<td>58</td>
</tr>
<tr>
<td>5.3</td>
<td>Cluster Mapping</td>
<td>59</td>
</tr>
<tr>
<td>5.4</td>
<td>Evaluation of Performance Sheet</td>
<td>59</td>
</tr>
<tr>
<td>6.</td>
<td><strong>Results</strong></td>
<td>64</td>
</tr>
<tr>
<td>7.</td>
<td><strong>Summary</strong></td>
<td>72</td>
</tr>
</tbody>
</table>

Appendix

References
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Overview of the project</td>
</tr>
<tr>
<td>2</td>
<td>General Structure of OCR System</td>
</tr>
<tr>
<td>3</td>
<td>OCR System construction</td>
</tr>
<tr>
<td>4</td>
<td>Machine-Printed and handwritten capital 'O'</td>
</tr>
<tr>
<td>5</td>
<td>an example for curved fitted baseline</td>
</tr>
<tr>
<td>6</td>
<td>A fixed-pitch chopped word</td>
</tr>
<tr>
<td>7</td>
<td>Difficult word spacing</td>
</tr>
<tr>
<td>8</td>
<td>Candidate chop points</td>
</tr>
<tr>
<td>9</td>
<td>An easily recognised word</td>
</tr>
<tr>
<td>10</td>
<td>(a) Pristine ‘h’, (b) Broken ‘h’, (c) Features matched to prototypes</td>
</tr>
<tr>
<td>11</td>
<td>Baseline and moment normalized letters</td>
</tr>
<tr>
<td>12</td>
<td>Logo of OpenCV</td>
</tr>
<tr>
<td>13</td>
<td>Representing text as a feature vector</td>
</tr>
<tr>
<td>14</td>
<td>Three well-separated clusters of 2-dimensional points</td>
</tr>
<tr>
<td>15</td>
<td>Centre based clusters</td>
</tr>
<tr>
<td>16</td>
<td>Contiguous Clusters</td>
</tr>
<tr>
<td>17</td>
<td>Density based clusters</td>
</tr>
<tr>
<td>18</td>
<td>Machine Learning Task</td>
</tr>
<tr>
<td>19</td>
<td>Types of machine learning</td>
</tr>
<tr>
<td>20</td>
<td>Projection of data in MLP</td>
</tr>
<tr>
<td>21</td>
<td>Radial Basis Function Network</td>
</tr>
<tr>
<td>22</td>
<td>Linearity separable data is shown in frame A, B, C and D show possible valid lines separating two classes of data</td>
</tr>
<tr>
<td>23</td>
<td>Maximize margin</td>
</tr>
<tr>
<td>24</td>
<td>Support Vector Machine model</td>
</tr>
<tr>
<td>25</td>
<td>K-Means algorithm</td>
</tr>
<tr>
<td>26</td>
<td>An insight into the clustering algorithm</td>
</tr>
<tr>
<td>27</td>
<td>Concept architecture</td>
</tr>
<tr>
<td>28</td>
<td>OCR Detection</td>
</tr>
<tr>
<td>29</td>
<td>Recognition system using SVM</td>
</tr>
<tr>
<td>30</td>
<td>H3 separated two classes with better margin than H1 and H2</td>
</tr>
<tr>
<td>31</td>
<td>Example of Cluster Connectivity</td>
</tr>
<tr>
<td>32</td>
<td>General Classification Segments</td>
</tr>
<tr>
<td>33</td>
<td>Data analysis of JNTUK University</td>
</tr>
<tr>
<td>34</td>
<td>Architecture</td>
</tr>
<tr>
<td>35</td>
<td>Sample Image of the first page</td>
</tr>
<tr>
<td>36</td>
<td>Sample Performance Sheet</td>
</tr>
<tr>
<td>37</td>
<td>OCR development steps</td>
</tr>
<tr>
<td>38</td>
<td>Training data of first page from LIBSVM</td>
</tr>
<tr>
<td>39</td>
<td>Training data of performance sheet from LIBSVM</td>
</tr>
<tr>
<td>40</td>
<td>Comparison of SVM and cluster approaches</td>
</tr>
</tbody>
</table>
Figure 41 Main page to login .......................................................... 66
Figure 42 Output with new split rules........................................... 67
Figure 43 Editing the output form new split rules ......................... 68
Figure 44 First page output .......................................................... 69
Figure 45 Output of performance sheet ....................................... 70
Figure 46 Complete processing steps ......................................... 71
1. Introduction

1.1. Motivation

There was an increase in the number of applications for a master degree program with the growth in time. It takes huge time to process all the application documents of each and every applicant manually and requires a high volume of the workforce. This can be reduced if automation is used for this process. In any case, before that, an analysis of the complete strides required in preparing was precisely the automation must be utilized to diminish the time and workforces must be finished. The application process for the applicant is actually participating in several steps. First, the applicant sends the complete scanned documents to the uni-assist; from there the applications are received by the student assistant team at the particular university to which the applicant had applied, and then they are sent to the individual departments. At the individual sections, the individual applications will be handled by leading an intensive study to know whether the applicant by their past capabilities scopes to satisfy the prerequisites of further study system to which they have applied. What's more, by considering the required points of interest of the applicant without investigating every single report, and to pack the information and diminish the preparing time for the specific division, by this postulation extend a solitary web apparatus is being produced that can procedure the application which is much dependable in the basic leadership procedure of application.

![Image of application process]

*Figure 1 Overview of the project*

Manually the processing of the application is, first we will receive a pdf document from the uni-assist containing each and every scanned document submitted by the applicant. From these documents, the specified complete details of the applicant that may be matched to the ASE program for further studies are going to be manually entered into the database by the auditors. In the event that the person crosses particular utmost focuses, then a confirmation in ASE is given to the applicant. There are some net applications for the process the applications during this approach. For any longer examination, all the data received from the student and uni-assist is stored. As the range of applications is increasing, the method needs much time, a large memory to store the information and a large number of auditors. From the uni-assist, we get a pdf document containing the first page, consists of the complete personal details of the applicant. In the university department, it is not necessary to consider full details of the applicant, and some of the data is in multiple entries on the first page. So by considering the selected keywords and to compress the data by reducing the space and making easier to the department portal a web application is developed. Additionally, I have done to work on the
performance sheet of the current student who is studying in the university. If a student changes their decision to modify the course by leaving the previous course. So, in the respective department on the need to check the performance sheet of previous study course and evaluated by undergoing different processes based on the subjects they had cleared and those which are suitable to enter the new course is evaluated. So, finally, the performance sheet evaluation is required to provide an indicator for the decision to change the study course, which is to reduce the time taking the process to the decision taking a person such as Dean of a particular department.

1.2. Problem Description

The issue faced in manual processing of the application is: Processing time of each application, multiple numbers of entries, manual data entry for further classification, each and every document has to be checked, and manual analysis of application numbers, less number of auditors, high data storage memory and less accuracy rate in classification. The main idea of this thesis project is to eradicate of these problems by introducing a tool which will alter pre-processing stage during this application method.

To automate the process, I have done the research work on many automation tools, pdf to text conversion tools and the others as explained in the state of the art. Finally came up with an idea to use OCR for pdf to text conversion and implement SVM and cluster-based approaches to classifying the data and the connectivity rule. And comparison both the results to urge the best classifier. With this approach we are able to alter pre-processing stage of the applying method and formulate the foundations of automatic detection of application information from OCR detection, reducing the time consumption for the applying method and process with the chosen keywords from the main points of the applicant the first-page area unit needed.

1.3. Project Information

This thesis project is done in cooperation with a professorship in Computer Engineering Department at Technische Universität Chemnitz. The main goal of the thesis is to create a tool that can formulate the rules for automatic detection of application data from OCR detection. To implement this, I used SVM and Cluster connectivity based approaches and finally compared both the data with the connectivity check and analyzing the result. The database ought to be outlined in a manner that it can be utilized for further advancements with new ways to deal with artificial intelligence. Finally, this tool can be further extended to other departments and study courses in the university.

1.4. Outlines of Thesis

Manual processing of the applications from uni-assist takes much time. We get the complete documents from the uni-assist to the university. By considering all the details from each and every document sent by the applicant, there exists a first page which consists the
important details of the applicant which is to be primarily checked by the University. This first page consists of the applicant personal details like name, date of birth, application number, birth place, gender, email id, previous studies credit points, university name and so on. First to analyze these details this pdf file of the first page is to be converted to the editable text file through OCR. For example in the case of the university. The name of the university is in three lines, to reduce the length of these lines to a single line and to analyze the correct data this tool is being developed. And this is applicable for each and every keyword which is selected from the first page. If we implement this type of tools with the keywords, then we can analyze this sort of certificates or the documents with very less time. To make this process more accurate, we try implementing SVM and Clustering methods. With these methods, we can accurately map the selected keywords from the first page and predict and train the data through the SVM and finally comparing with the Cluster connectivity. At some point we cannot get correct results with the SVM, so we need to compare with the cluster connectivity to analyze the result.

1.5. Document Layout

A complete document of the thesis is divided into five chapters. Each and every chapter handles different topics. Chapter 1 introduces the topic. Chapter 2 gives the background knowledge and research made on previously existing methods. It also briefs some detailed knowledge in the character recognition and machine learning methods that exist. Chapter 3 gives the information about the complete concept involved in this project. In this chapter, we discuss more the modules arranged to obtain the final result. Working on each and every module and their results that can be used in the further processing. Chapter 4 tells about the comparison results between SVM and Cluster connectivity methods. And finally, chapter 6 gives a summary of further implements and phases with developments that can be accomplished.
2. Basic Information

2.1. OCR (Optical Character Recognition)

An Optical Character Recognition has a place in the family of techniques performing the automatic identification [1]. This process can be used when the information or the data is in the form of readable to both humans and to the machine, and the alternative inputs cannot be predefined. When compared to other techniques for automatic identification, optical character recognition is unique as it doesn't require control of the procedure that delivers the data. It mainly deals with the problem of recognizing optically processed characters. The performance directly depends on upon the quality of the input documents. Optical Character Recognition is a field that is broadly utilized in pattern recognition, artificial intelligence, and Machine vision. The significance of OCR in today's applications can't be over-emphasized. The territory of OCR is turning into a fundamental piece of document scanners and is utilized as a part of numerous applications, for example, postal processing, script recognition, banking, security and language identification. Numerous OCR procedures have advanced overtime to achieve these assignments since the introduction of the first OCR system called “Reading Machine” in Germany by Tausheck in 1929, and an American patent was filed subsequently 1935. Later on, technology was improved and developed the OCR in early 1950. The research in this area has been ongoing for over half a century, and the outcomes have been astounding with favorable recognition rates for printed characters surpassing 99% with critical upgrades in execution for handwritten cursive character recognition where recognition rates have surpassed the 90% imprint [2]. Documents are scanned using a scanner and are given to the OCR systems which identify the Characters in the scanned documents and converts them into ASCII data.

The OCR algorithms have been proposed in the literature depends on either character matching using the templates, image shapes, geometric properties and the invariant techniques based on shapes. In the template matching for the instance, a binary image of the characters is being stored in the database. When the given text image is captured the algorithm tries to segment the image into its component text line and extract each character from a given input. Now each extracted character is compared to all the templates stored in the database using the correlation to find a match. In a portion of the algorithms, the element vectors are extricated from the divided character utilizing an algorithm, for example, Local Discriminant Analysis (LDA), Local Binary Pattern (LPB), and Principal Component Analysis (PCA). These component vectors are typically used to prepare a classifier like Artificial Neural Network and Support Vector Machine (SVM) to perceive the character. For number recognition, each character was extracted by summing all the pixels values of the binarized plate number along the rows. Zero values boundaries are recognized as the end of the character, and simultaneously a template matching is used for the classification. In this approach after extracting all the characters in a line, an average character width is computed. And any character whose width is greater than average is sent to clustering algorithm which detects multiple characters and separates them.
2.1.1. Construction of OCR System

The typical OCR system consists of three logical components [3]:

- Image Scanner
- OCR Software and Hardware
- Output Interface

The Scanner optically captures text images to be recognized. These text images are managed with OCR software and hardware. The process involves three operations: document analysis by extracting individual character images, recognizing these images based on the shape and the contextual processing either to correct the misclassifications made by the recognition algorithm. The detector consists of an array of elements, each of them transforms instance light into an analog signal. These simple signs are then changed over to a picture. Filtering is performed by the indicator and the movement of the content item with deference to the identifier. After a picture is caught, the report transport expels the record from the scanning field.

The types of OCR frameworks are portrayed by specific features. The sorts of OCR are the Auxiliary sort, feature sort, and neural system sort. A few usage utilize State Vector Machine

Figure 2 General Structure of OCR System
(SVM) alongside the wavelets as the info data for the OCR process. In order to drive the wavelets, these strategies include numerous complex scientific counts. These, thus, utilize framework increases and summations. One more imperative thing to be noted is, the SVM basically requires time for taking distinctive info yield combines and computing the connection between those information yield sets. Then again, frameworks which don't utilize a preparation instrument or memory based preparing or recognition are likewise accessible. A couple of these frameworks utilizes fluffy rationale, and histogram sort ranges weight discovery of the letters in order. Another sort is feature recognition sort OCRs; distinctive features are separated from the letter set present in the info. This technique is worthwhile as far as memory use and calculations, as constrained features are sufficient to recognize the character.

Underneath we talk about these distinctive procedures and characterize OCR's position among them. Automatic ID, The customary method for entering data into a computer, is through the console. Be that as it may, this is not generally the best nor the most productive arrangement. By and large, automatic distinguishing proof might be an option [1]. Different technologies for automatically recognizable proof exist, and they cover requirements for various regions of application. Beneath a brief outline of the diverse technologies and their applications are Speech recognition, Radio frequency, Vision systems, Magnetic stripe, Bar code, Magnetic ink, Optical mark reading.

The OCR system is comprised of several parts, but it is divided into four major parts:

- Pre-processing
- Self-learning
- Recognition
- Post processing

![Figure 3 OCR System construction](image-url)
2.1.1.1. Pre-processing

The pre-processing method takes the input image obtained from the input supply like a scanner or the other source and transfers the first text image into either a picture file or a binary matrix. Then the image undergoes a method referred to as text analysis. During this method, the text image is sectional into lines and characters. Thus on remake the ASCII document, the position of every character is recorded. Text analysis could include the following:

- **Extraction of character from an image**

  Extraction of character region from a picture is accomplished by exploitation the auxiliary data famed concerning the image to select the image properties that have a spare variation of the text region, and therefore the background regions to be a basis of separating one from another. The white area includes a typical signature that we tend to use to section the histogram and thereby the lines of text [4]. Also, we are able to recognize the expected character size of the text, or alternative spatial organization or layout of the text. Choosing a mask in order that morphology operations grow those characters for connecting them to make word regions without bridging the area that is present between words or between lines. Using the nearby Fourier cover or surface veils, the regions of high spatial frequencies and consequently literary stage districts will be recognized.

- **Segmentation of the image into text and background**

  While some OCR algorithms work with grayscale pictures, numerous believers their contribution to paired picture designs amid the early phases of handling. At the point when a picture locale is containing content, whether single word districts or entire pieces of content, is given, the objective of the stage recognizes the picture pixels which have a place with the content and the pixels which have a place with the foundation. Thresholding of the dark level picture is that the most generally utilized methodologies. we can choose the edge esteem from various perspectives like misuse auxiliary data in regards to the picture, abuse factual methods to choose a worldwide limit that "best" groups the picture into 2 classes and by abuse required measures figured inside the area of each pixel to see the best neighborhood edge. Practically, it's determined that the local adaptive thresholds manufacture the most effective segmentation results.

2.1.1.2. Self-Learning

Each present day OCR apparatus has this extraordinary property of self-adapting in this way on improving the data at whatever point an obscure character string is acquired. It is perceived in view of the data beforehand inbuilt. This information contains most vital feature regarding the character that is already best-known. As additional and more new characters are met, the recognition ability is often enhanced if the information is capable of self-expanding.
2.1.1.3. Recognition

Of all the parts of the system, the Recognition part is the major one. It separates info character elements and contrasts them and the components that were recorded in the database utilized by the acknowledgment. The two most important essential components in a character recognition algorithm are the feature extraction and the classifier. In the element analysis, it decides the descriptors or the list of capabilities used to portray every one of the characters. In the event that given a character picture, the element extractor determines the elements that the character have. Whether the Characteristics would coordinate or about coordinated, the useful characters will be requested under those populates amid which each of the ones in everything about characters craves these typical components. The last step during this stage is final classification. During this stage, the character hold on to this block is written into the result, and therefore the different characters represent broad classification. The template matching or the matrix matching is one of the most common methods. In the template matching, individual image pixels are used as features. The classification is performed by contrasting an information character picture and an arrangement of formats from every character class. Each comparison results in the similarity measure between the character of entry and the template. After all the templates are compared with the ascertained character image and therefore the characters identity is assigned because the identity of the foremost comparable pattern.

The structural classification methods use structural features, and the decision rules to classify characters. Structural features can be defined in terms of character strokes, character holes or the other character attributes. For instance, the letter ‘P’ could also be represented as a vertical stroke with a loop connected to the higher right aspect. For a character picture entering, the auxiliary features are extricated, and a tenet based framework is connected to classify the character.

![Figure 4 Machine-Printed and handwritten capital 'O'.](image)

The classification is performed by contrasting an information character picture and an arrangement of formats from every character class. These recognizers can use the pixel-based features or structural features. Some of the examples are functional discriminant classifiers, Bayesian classifiers, artificial neural networks (ANNs) and the template matches. The character misclassification begins from two fundamental sources: low quality character pictures and poor
prejudicial capacity. The document poor quality, image scanning and pre-processing can all degrade performance by yielding poor-quality characters. The character acknowledgment technique might not have been prepared for an appropriate reaction to the character bringing about a mistake. This sort of blunder sources hard to overcome as the acknowledgment techniques may have constraints and all conceivable character pictures can't in any way, shape or form be considered in preparing the classifier. The recognition rates for the machine printed characters can reach over 99%, but the handwritten character recognition rates are typically low because it differs with the handwriting. In fig:4 a few case of machine printed and handwritten capital O's can be effectively mistaken for the numeral 0, and the quantity of various styles of capital O's shows the challenges of the recognizers must adapt.

**Feature Detection:** For any OCR, feature detection and classification are the centres. Since the appearance of OCR many feature detection techniques are used. First, the total character is found as a feature by template matching, and so the sub-features of the character are sought-after. The algorithmic rule found the boundary outlines, the character skeleton or medial axis, the Fourier or wave coefficients of the spatial pattern, numerous moments each spatial and gray level, and topological properties equivalent to the quantity of holes during a pattern. On the other hand, the algorithmic rule designers were interested in those features that exhibit no variation with image changes, like translation, rotation, contrast and color. For detection of options, statistical correlation, boundary following, region thinking, mathematical morphology, and additionally entirely unexpected changes like the Fourier change, composition veil, spatial example operations, and topological algorithms are utilized. Before character segmentation, feature location systems are connected. Researchers on the search for new techniques for finding characters in the text without prior character segmentation are increasing rapidly.

**2.1.1.4. Post Processing**

The prime motivation behind Post-preparing stage is to make important revisions if any required. The ASCII content comprises both perceived and rejected character, and the character requires further altering and adjustment. So the ensuing character that is obtained when recognition stage has got to be moved either to the would like a (correct) list or into the rejecting list.

**OCR Implementation Steps**

The template-matching algorithm implements the following steps [5]:

- Firstly, the character image from the detected string is selected.
- After that, the image to the size of the first template is rescaled.
- After rescaling the image to the size of the first template (original) image, the matching metric is computed.
- Then the highest match found is stored. If the image is not matched, repeat the third step again.
- The index of the best match is stored as the familiar character.
2.2. Tesseract

Tesseract is an open-source OCR motor that was created at HP somewhere around 1984 and 1994 [6]. It showed up from no place for the 1995 UNLV Annual Test of OCR Accuracy, demonstrated brilliantly with its outcomes, and then vanished back under the same shroud of mystery under which it had been created. Presently surprisingly, the complete points of interest of the engineering and the algorithms can be uncovered. Tesseract started as a Ph.D. research venture in HP Labs, Bristol, and picked up force as a conceivable programming and/or equipment add-on for HP's line of flatbed scanners. After a joint undertaking between HP Labs Bristol and HP's scanner division in Colorado, Tesseract had a noteworthy lead in precision over the business motors, however, did not turn into an item. The following phase of its advancement was back in HP Labs Bristol as an examination of OCR for pressure. Work focused more on enhancing dismissal productivity than on base-level precision. Toward the end of this anticipate, toward the end of 1994, improvement stopped completely. The motor was sent to UNLV for the 1995 Annual Test of OCR Accuracy, where it demonstrated its value against the business motors of the time. In late 2005, HP discharged Tesseract for open source.

Tesseract needs a small amount of pre-processing to enhance the OCR results; pictures have to be compelled to be scaled appropriately, have the maximum amount image contrast as attainable, and have horizontally-aligned text it's currently accessible at http://code.google.com/p/tesseract-ocr.

2.2.1. Architecture

The Later H.P. research lab developed layout analysis technology severally. The technology was employed in the product. (And so not free to open-source) because of this technology developed by H.P., Tesseract ne'er met the requirement of its own page layout analysis. Tesseract assumes that its input could be a binary image with optional plane figure text regions are being outlined.

The Later H.P. workplace developed layout analysis technology severally. The technology was used in the product. (And thus not discharged to open-source) because of this technology developed by H.P., Tesseract ne'er met the need of its own page layout analysis. Tesseract assumes that its input may well be a binary image with optional figure text regions unit being made public. the process follows a standard stepwise pipeline, but a variety of the stages were strange and settled very new in those days, and doubtless keep therefore even presently. the first step of the method may well be a connected part analysis. throughout this step, the outlines of the weather square measure keep. With relevance computation, the design decision was a chic one at the time. be that as it may, it had a noteworthy preferred standpoint: by assessing the settling of layouts, and additionally the vary of child and relation outlines, the detection of inverse text became straightforward, and recognition of the text became very easy in recognition of black-on-white text. Tesseract is maybe the first OCR engine that is capable of handling white-on-black text, therefore, a bit effort. At this stage, outlines unit created on, strictly by nesting, into Blobs [6]. Blobs square measure unionized into text lines, and analysis
of the lines and regions for mounted pitch or proportional text is finished. Supported the kind of character spacing, text lines unit broken into words. The character cells cut the mounted pitch text in real time. The proportional text is broken into words by mistreatment the definite areas and fuzzy areas.

At that point, recognition proceeds as a two-pass process. In the principal pass, recognition of every word, thus, is endeavored. Every word that is acceptable is passed to a versatile classifier as a preparation data. At that point the versatile classifier gets a chance for recognizing the content let down the page with still more exactness. Since the flexible classifier may have learned something supportive past the point where it is conceivable to make a pledge near the most elevated purpose of the page, a second nonchalance is run the page. In the second pass, the words which were not appropriately recognized are made to recognize at the end of the day, in this way prompt more prominent exactness of recognition. The last stage determines fluffy spaces, and checks elective theory for the x-tallness to find little top content.

2.2.2. Line and Word Finding

2.2.2.1. Line finding

There are additional publications earlier on-line finding; that is one amongst the main components of Tesseract. The most application of this rule which is meant for Tesseract is to classify skew pages. This method is more involved regarding line construction and blob fitting. The line finding rule is one in every of the few components of Tesseract that was printed earlier. This rule is meant for recognizing a skew page without having to de-skew. This, in the end, ensured the picture quality. Blob, blob sifting, and line development are the key segments of the technique. Since the generally uniform content size, the drop-tops, and vertically touching characters are evacuated by a simple imprint stature channel with a supposition of page design analysis. The text within the region is judged by the peak of the median. As a result of this, filtering out the blobs that are petite then some portion of the median height, maybe, discriminating marks, punctuation, and noise is safe.

The filtered blobs that area unit a lot of precise to suit a model of non-covering, parallel and slanting lines. In the event that the blobs are sorted and processed by the x-coordinate, skew gets to be viable in some conditions like assigning the blobs to a singular text line, across the page throughout the method of tracking the slope, with the heavily subsided danger of distribution to associate degree incorrect text line within the existence of skew. The final step of the line creation process to associate blobs that cover by at any rate half on a level plane, assembling diacritical imprints with the right base and effectively relating parts of some broken characters[6].

2.2.2.2. Baseline Finding

Once the text lines are detected, the baselines are fixed in additional accurately using a quadratic spline. This was very 1st finished an OCR system and enabled Tesseract to treat pages with recurvate baselines, that are a bit familiar object in scanning, and not simply a kind of book.
The blobs are assigned into groups with a sensibly nonstop removal for the first straight standard. This can be. However, the baselines are fitted. A quadratic spline is fitted to the segment that is observed to be the principal crowded (thought to be the standard) by a minimum square fit. The quadratic spline has every favorable position and a disadvantage. The advantage being “this calculation within reason stable” and therefore the disadvantage being “discontinuities will arise once multiple spline segments are needed”. An additional regular cubic spline may work much better.

The line of content with a right standard, descender line, mean line and ascender line are represented in the above fig. Every one of the lines appeared in the picture are "parallel" as their y detachment may be a constant for the complete length. Also, these lines are slightly curved. It may be determined within the image that the ascender line is cyan (prints as lightweight gray), and also the black line present concerning this cyan line is very. A profound perception of these lines makes it clear that the cyan (printed dim) line is bent with respect to the straight dark line above it.

2.2.2.3. Fixed Pitch Recognition and Chopping

The content lines will be tried by Tesseract for choosing whether they are altered pitch content. Then again, Tesseract cop's the words into characters misuse the pitch and additionally disables the chopper and associated on these words for the word recognition step. The pitch is also used to measure the height of the character.

A sample example of a fixed-pitch word is proved in below fig.

Figure 5 an example for curved fitted baseline

A fixed-pitch chopped word

2.2.2.4. Proportionate word finding

Non-fixed-pitch or proportional text spacing may be an extremely non-trivial task. Observe the Fig.3 that illustrates some typical issues. The gap present between the tens and units of 11.9% is equivalent in size to the general space. However, this is frequently really extra contrasted with the kernel space present amongst "erated" and "junk". No horizontal gap exists
in any respect between the bounding boxes of 'of and financial'. Many of such issues are solved by Tesseract by activity gaps during a limited vertical range existing between the baseline and mean line. At this stage, the areas that are near to the edge are created fuzzy, for creating a final decision on word recognition.

of 9.5% annually while the Federated junk fund returned 11.9%

Figure 7 Difficult word spacing

2.2.3. Word Recognition

For any character recognition motor, the recognition strategy needs to decide the methodology a word ought to be divided into characters. To begin with, the classification of the underlying segmentation yield from line finding is finished. The rest of the part of the word recognition step is relevant to non-settled pitch message as it were.

2.2.3.1. Chopping Joined Characters

While the results of a word from Linguistic Analysis that are a terribly\} position to be mentioned terribly shortly is disappointing, Tesseract marks an effort for rising the result by chopping the blob with worst confidence from the character classifier. Applier chops points are found from saclike peaks of a figure approximation of the defined, and might need either altogether utterly totally different vertex opposite, or a line half. So on separately joined characters from the ASCII code set successfully, up to several pairs of chop points is additionally needed.

Figure 8 Candidate chop points

Above figure is a typical example showing a set of candidate chop points with the arrows and a selected chop as a line cross the outline where the character ‘r’ touches the
character’. The success of Tesseract was owing to a necessary part referred to as character classifier that would simply recognize is broken characters.

### 2.2.3.2. Associating the Broken Characters

When the potential chops are exhausted, if the word is still not clear, then it is given to the associate. The associate makes an A* (best first) hunt of the division diagram of conceivable blends of the maximally hacked blobs into the candidate characters. It should be possible without building the division diagram however rather keeps up a hash table of went to states. The A* search proceeds by pulling the candidate new states from a priority queue and evaluating them by classifying the unclassified combinations of fragments. It might be contended this completely hack then-relate methodology is, best case scenario wasteful, at the very least at risk to miss the basic cleaves. The main advantage of chop-then-associate scheme simplifies the data structures that would require maintaining the full segmentation graph.

![Figure 9 An easily recognised word](image)

When A* segmentation search was first implemented in about 1989, Tesseract’s accuracy on the broken characters is well ahead of the commercial engines of the day. From fig 9, an essential part of that success was the character classifier that can be easily recognized broken characters.

### 2.2.4. Static Character Classifier

#### 2.2.4.1. Features

The earlier version of Tesseract used topological options that are developed from the work of Shillman. In spite of the very fact that these components are autonomous of text style and size, they're not sturdy to the issues being knowledgeable about in real time images. The very next thought that may crack to mend this was to create use of segments of the two-dimensional figure approximation as features. Unfortunately, this approach is additionally not

![Figure 10 (a) Pristine ‘h’, (b) Broken ‘h’, (c) Features matched to prototypes](image)
strong to the broken characters. As an example, in below figure 10 (b), it is often discovered that the correct aspect of the shaft is on two main items, however in figure 10 (a) simply a single piece is detected.

Then an abrupt and dramatic solution came up. It’s the concept that the options within the unknown and therefore the options within the training information each needn't be an equivalent. Within the training method, the segments of a plane figure approximation are used for options, whereas within the case of recognition, extraction of options of a small, fixed length of the outline is completed and that they are matched many-to-one against the clustered image options of the training information. The short, thick lines found inside the above figure, are the choices that are separated from the obscure, though the slim and long lines inside the same figure are the grouped sections of the plane figure estimation utilized as models. One model bridging the two items is entirely unmatched. All the prototypes and each feature are compatible except for the three features on one aspect and two on the opposite. This exemplifies that the method of little features matching with the large prototypes is definitely ready to alter recognition of the pictures that are broken. Its prime disadvantage was the estimation of registering the crevice between partner obscure, and an image is entirely high.

In this manner, the components that are extricated from the obscure are 3-dimensional, i.e., has a x position, y position and edge with generally 50-100 elements amid a character, while the model elements are 4-dimensional, i.e., x position, y position, point and length with as a rule 10-20 highlights amid a model setup.

2.2.4.2. Classification

Classification may be a two-step process; whereby the primary step is that the creation of a brief list of character categories that the unknown would possibly match. This can be done by a category pruner. From a coarsely quantal 3-dimensional operation table, each feature can fetch a bit-vector of the categories that it'd match. These bit-vectors fetched are then summed over all the options. The categories with the very best counts (after correcting for expected variety of features) become the list for the next step.

Every feature of the unknown searches a small amount vector of prototypes of the given category that it'd match with. Then computation of the particular similarity between them is completed. Each paradigm character category is diagrammatical by a logical SOP (sum-of-product) expression with every term. It’s known as a configuration. A record of the overall similarity proof {of each} feature in every configuration, and additionally in every paradigm is maintained by the gap calculation method. The best-combined distance that is calculated from the total of feature and paradigm proof is that the best overall the saved configurations of the category.
2.2.4.3. Training Data

The classifier is capable of recognizing the broken characters with great ease. So, the classifier wasn't trained on broken characters. The classifier was obviously prepared over a few specimens, i.e., 20 tests of 94 characters from 8 text styles in a single size, however with four qualities (ordinary, intense, italic, striking italic). Of these mean a complete of 60160 preparing tests. This is frequently a major difference contrasted with various printed classifiers, similar to the Calera classifier that is prepared in more than 1,000,000 examples and Baird's 100-textual style classifier that is prepared on 1175000 preparing tests.

2.2.5. Adaptive Classifier

It was proposed and even shown that the usage of an adaptative classifier would advantage the OCR motors. The static classifier should be sensible at generalizing to any quite font. As a result of that the power of the classifier to discriminate between completely different characters or between characters and non-characters weaken. Along these lines, for understanding a bigger separation among each archive, wherever a less number of textual styles are existing, an extra-textual style delicate versatile classifier prepared by the consequences of the static classifier is regularly utilized.

Template classifier isn't employed by Tesseract. However, it uses the similar options and classifiers of the static classifier. The static classifier can standardize the characters by the centroid (first minutes) for position and second minutes for anisotropic size standardization.

The standardization of baseline/x-tallness will can without much of a stretch separate upper and lower case characters [6]. It will likewise enhance the freedom to clamor spikes. The real favorable position lying behind the character minute standardization is evacuated of textual style perspective proportion and some level of text style stroke width. Not only that, however, it additionally makes the recognition of sub and superscripts easier, however, wants an extra classifier's feature for marking the distinction between some upper and lower-case letter characters. Three letters in baseline/x-height normalized type and moment, normalized type are illustrated in below figure.

![Figure 11 Baseline and moment normalized letters](image-url)
2.3. OpenCV

OpenCV is an open source (see http://opensource.org) computer vision library offered from http://SourceForge.net/projects/opencvlibrary. Mostly written in C and C++ and runs under different types of OS like Linux, Windows, and mac OS X. There’s an active development on interfaces for Python, Ruby, Matlab, and different languages [7].

The OpenCV library began as a research project at Intel in 1998. It has been accessible since 2000 under the BSD open source license. OpenCV was designed for computational efficiency and with a strong focus on real-time applications. It is a set of normally used functions that perform operations relating to computer vision. OpenCV has been natively written in C/C++, however, has wrappers for Python, Java, and any JVM language, that is meant to make the Java bytecode, like Scala and Clojure. Since most of the Android app development is finished in C++/Java, OpenCV has additionally been ported as associate SDK that developers will use to implement it in their apps and create them vision enabled.

Initially developed by Intel, OpenCV is a free cross-platform library for real-time image process for all things relating to Computer Vision. OpenCV is aimed at providing the tools required to unravel computer vision issues. It contains a combination of low-level image-processing functions and high-level algorithms like face detection, pedestrian detection, feature matching, and tracking.

One of OpenCV’s goals is to supply a simple-to-use computer vision infrastructure that helps individuals build fairly refined vision applications quickly. The OpenCV library contains over five hundred functions that span several areas in vision, as well as industrial plant product examination, medical imaging, security, computer program, camera standardization, stereo vision, and artificial intelligence. As a result of computer vision and machine learning usually go hand-in-hand? OpenCV conjointly contains a full, all-purpose Machine Learning Library (MLL). This sub-library is concentrated on applied mathematics pattern recognition and cluster. The MLL is extremely helpful for the vision tasks that square measure at the core of OpenCV’s mission.

Figure 12 Logo of OpenCV
2.4. Support Vector Machines

Support Vector Machine introduced by Vladimir Vapnik in 1960’s. SVM) is a supervised machine learning calculation which can be utilized for both classification and relapse challenges. Notwithstanding, it's to a great extent used in classification issues. Support Vectors are just the directions of individual perception [8]. It is a boondocks which best isolates the two classes (hyper-plane/line). In machine learning, support vector machines are supervised learning models with related learning calculations that investigate information utilized for classification and multivariate examination. Given a gathering of preparing illustrations, each set apart to belong to 1 of 2 classes, a SVM preparing decide manufactures a model that relegates new case into one class or the inverse, making it a non-probabilistic paired direct classifier. a SVM model might be a representation of the case as focused in the region, mapped all together that the examples of the different classes are isolated by a straightforward crevice that is as wide as would be prudent. New cases are then mapped into that exceptionally same territory and anticipated to have a place with a class in light of that part of the crevice they fall on. Furthermore, to performing direct classification, SVMs will with proficiency play out a non-straight classification misuse what's known as the part trap, certainly mapping their inputs into high-dimensional component regions. When data don't appear to be named, supervised learning isn't conceivable, and an unsupervised learning methodology is required, that tries to discover regular bunching of the data to gatherings, thus outline information to those shaped gatherings. The cluster rule that provides an improvement to the support vector machines is termed support vector clustering and is commonly utilized in industrial applications either once information isn't tagged or once just some information is labeled as a pre-processing for a classification.

SVM in Text Classification:

The objective of content arrangement is that the classification of documents into a set scope of predefined classes. Each document d is in different, particularly one, or no classification at all. Using machine learning, the goal is to learn classifiers from illustrations which do the class assignments naturally. To facilitate effective and efficient learning, every class is treated as a separate binary classification drawback. Each such issues answers the inquiry, whether a document ought to be doled out to a specific classification or not.

Representing text:

The representation of an issue strongly affects the speculation exactness of a learning framework [9]. Documents, which usually are strings of characters, need to be transformed into a representation appropriate for the training algorithmic rule and therefore the classification task. The researchers recommend that word stems work well as illustration units which their ordering in an exceeding document is of minor importance for several tasks. The word stem is
gotten from the event type of a nutshell by expelling the case and flection data [Porter, 1980].
To illustrate "computes," "computing", and "computer" are all mapped to a proportional stem "computer." The expressions "word" and "word stem" are utilized synonymously as a part of the accompanying. This winds up in a trait esteem delineation of content. Each particular word \( w_i \) relates to a component with \((w_i, d)\), the quantity of times word \( w_i \) happens in the document \( d \), as its worth. The following figure shows the feature vector for a particular document.

![Figure 13 Representing text as a feature vector](image)

To avoid unnecessarily massive feature vectors, words area unit thought of as options as long as they occur within the training data a minimum of three times and if they're not “stop-words” (like “and,” “or,” etc.) [10]. In view of this fundamental representation, it is realized that scaling the measurements of the element vector with their converse document recurrence IDF \((w_i)\) [Salton and Buckley, 1988] prompts an enhanced execution. IDF \((w_i)\) can be ascertained from the document recurrence DF \((w_i)\), which is the quantity of documents the word \( w_i \) happens in.

\[ IDF(w_i) = \log \left( \frac{n}{DF(w_i)} \right) \]

Here, \( n \) is the total number pf training documents. In the event that the reverse document recurrence is low on the off chance that it happens in numerous documents and is most noteworthy if the word happens in one and only. To digest from various document lengths, every document highlight vector \( d_i \) is standardized to unit length.

### 2.5. Cluster

Clustering can be seen as the most fundamental unsupervised learning issue; thusly, like each other issue of this kind, it oversees finding a structure in a gathering of unlabeled information. A free meaning of clustering could be "the way toward sorting out objects into
gatherings whose individuals are comparative somehow”. A cluster is, accordingly, an accumulation of objects which are "comparative" to them and are "disparate" to the objects having a place with different clusters. Clustering does not utilize beforehand doled out class names, with the exception of maybe for confirmation of how well the clustering functioned. Consequently, cluster investigation is unmistakable from example acknowledgment or the regions of measurements know as discriminant examination and choice examination, which look to discover rules for classifying objects given an arrangement of pre-classified objects. Cluster investigation can be helpful either specifically or as a preparatory method for discovering classes, there is significantly more to these zones than cluster examination. For instance, the choice of what features to utilize when speaking of objects is a center movement of fields, for example, design acknowledgment [11]. Cluster examination normally takes the components as given and proceeds to start there a couple working implications of bunches which are generally used are according to the ensuing:

**Well-Separated Cluster**: A cluster is an arrangement of focuses such that at whatever time in a cluster is closer (or more comparative) to each other perspective in the cluster than to any point not in the cluster. Now and then a limit is utilized to indicate that every one of the focuses in a cluster must be adequately close (or comparative) to each other.

**Figure 14 Three well-separated clusters of 2-dimensional points**

**Centre-based Cluster**: A cluster is an arrangement of objects such that an object in a cluster is nearer (more comparative) to the "middle" of a cluster than to the focal point of some other cluster. The focal point of a cluster is frequently a centroid, the normal of the considerable number of focuses in the cluster, or a medoid, the most "illustrative" purpose of a cluster.

**Figure 15 Centre based clusters**
Continuous Cluster (Nearest neighbor or Transitive Clustering): A cluster is an arrangement of focused such that a point in a cluster is closer (or more comparative) to one or more different properties in the cluster than to any point not in the cluster.

![Figure 16 Contiguous Clusters](image)

**Density-based definition:** A cluster is a thick district of focuses, which is isolated by low-thickness areas, from different locales of high thickness. This definition is all the more frequently utilized when the clusters are sporadic or entwined, and when clamor and anomalies are available. Here assumption of highest density only, may mention, so may define density-based local coverage check (empty spaces of certain density only allowed) or connection by outlining (one (dense) neighbour in every position) more in direction of outlining with some tolerance.

![Figure 17 Density based clusters](image)

**Similarity-based Cluster:** A cluster is an arrangement of objects that are "comparative", and objects in different clusters are not "comparative." A minor departure from this is to characterize a cluster as an arrangement of focuses that together make a district with a uniform neighborhood property, e.g., thickness or shape.

**Common Proximity Measures in Cluster Distance Measures:** The most ordinarily utilized nearness measure, at any rate for proportion (scales with an outright 0) is the Minkowski metric, because real distance included would go more for focus instead of speculation in Euclidean. It is defined as

\[
P_{ij} = \left( \sum_{k=1}^{d} |x_{ik} - x_{jk}|^r \right)^{\frac{1}{r}}
\]

Where, \( r \) is a parameter, \( d \) is the dimensionality of the information object, and \( x_{ik} \) and \( x_{jk} \) are, separately, the \( k^{th} \) components of the \( i^{th} \) and \( j^{th} \) objects, \( x_i \) and \( x_j \).

The following is a list of the common Minkowski distances for specific values of \( r \).

1) \( r = 1 \). City block (Manhattan, taxicab, L1 norm) distance.
   A typical case of this is the Hamming separation, which is only the quantity of bits that are diverse between two twofold vectors.
2) \( r = 2 \). Euclidean distance. The most widely recognized measure of the separation between two focuses.

3) \( r \to \infty \). “Supremum” (Lmax norm, L\( \infty \) norm) distance. This is the maximum difference between any components of the vectors.

**Ordinal Measures:** Another regular kind of vicinity measure is determined by positioning the separations between sets of focuses from 1 to \( m \ast (m - 1)/2 \). This sort of measure can be utilized with most sorts of information and is regularly utilized with some of the various leveled clustering algorithms.

**Similarity Measures Between Binary Vectors:** There are numerous measures of similitude between twofold vectors. These measures are alluded to as closeness coefficients and have values somewhere around 0 and 1. An estimation of 1 shows that the two vectors are completely comparable, while an estimation of 0 demonstrates that the vectors are not under any condition comparable. There are numerous reasons for why one coefficient is superior to another in particular examples, yet we will specify just a couple. The comparison of two paired vectors, \( p \), and \( q \), prompts four amounts:

- \( M_{01} \) = the number of points where \( p \) was 0 and \( q \) was 1
- \( M_{10} \) = the number of points where \( p \) was 1 and \( q \) was 0
- \( M_{00} \) = the number of points where \( p \) was 0 and \( q \) was 0
- \( M_{11} \) = the number of points where \( p \) was 1 and \( q \) was 1

The most straightforward closeness coefficient is the basic coordinating coefficient

\[
SMC = \frac{M_{11} + M_{00}}{M_{01} + M_{10} + M_{11} + M_{00}}
\]

Another generally used measure is the Jaccard coefficient.

\[
J = \frac{M_{11}}{M_{01} + M_{10} + M_{11}}
\]

Reasonably, SMC compares similitude with the aggregate number of matches, while J considers just matches on 1’s to be critical. It is necessary to understand that there are circumstances in which both measures are more proper [11]. For instance, consider the accompanying sets of vectors and their straightforward coordinating and Jaccard likeness factors.

\[
a = 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ SMC = .8 \\
b = 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 1 \ J = 0
\]

If the vectors are assumed as the answers for the true or false criteria, if the 0-0 and 1-1 matches then the two answer are assumed to be similar. This criteria may be assumed for the grading points as well.

**Cosine Measure:** For reasons for data retrieval, e.g., as in web crawlers, documents are regularly put away as vectors, where every characteristic speaks to the recurrence of which a
specific term (word) happens in the document. (It is a great deal than this, obviously, since certain basic words are disregarded and standardization is performed to represent diverse types of the same word, document length, and so on.) Such vectors can have thousands or countless qualities. On the off chance that we wish to look at the closeness of two documents, then we confront a circumstance much like that in the past segment, aside from that the vectors are not parallel vectors. Everything we require a measure like the Jaccard measure, which disregards 0-0 matches. The inspiration is that any two documents are liable to "not contain" a hefty portion of the same words, and in this way if 0-0 matches are numbered most documents will be very like most different documents. The cosine measure, characterized underneath, is the most widely recognized measure of document likeness. If \( d_1 \) and \( d_2 \) are two text vectors, then

\[
\cos(d_1, d_2) = \frac{d_1 \cdot d_2}{\|d_1\| \|d_2\|}
\]

Where \( \|d_2\| \) is the length of the vector \( d \).

**Time and Space Complexity:**

As the vectors are put away, the space prerequisites are essential \( O(mn) \), where \( m \) is the quantity of focuses and \( n \) is the quantity of properties. The time necessities are \( O(I \times K \times m \times n) \), where \( I \) is the quantity of emphases required for joining is. \( I \) is ordinarily little and can be effectively limited as most changes happen in the initial couple of emphases. In this way, K-means is straight in \( m \), the quantity of focuses, and is productive, and basic, the length of the quantity of clusters is fundamentally not as much as \( m \).
3. State of the Art

In this module, we discuss the summary of the basic and background knowledge we found in the research work for this thesis.

3.1. Text Mining

Most past examinations of information mining have concentrated on helper information, as social, worth based, and information stockroom information [12]. Practically speaking, a significantly higher measure of the accessible data is went down in a content database (or document database), which includes a huge gathering of documents from a few unique sources, similar to news articles, research papers, books, computerized libraries, email messages, and pages. Inferable from the grand climb in the measure of data open in electronic structure, for instance, electronic appropriations, and distinctive sorts of electronic records, email, and the Internet, there is a dubious and sudden rising in the content database. Nowadays, all the data on government, industry, business and different organizations are put away electronically, i.e., as a content database.

When data is transferred to the database, the majority of this data will be in an unstructured way. Lately, we discover much research is centered on semi-organized data. Moreover, data recuperation procedures, for instance, indexing methods, have been made to handle unstructured data.

Ordinary data retrieval methods get the chance to be insufficient for the relentlessly unfathomable measure of content data. Regularly, only a little partition of the various open documents will be pertinent giving the individual customer. Without perceiving what could be in the documents, it was hard to characterize feasible request for dissecting and getting required data from data. Clients need apparatuses to analyze diverse documents, rank the documents, or discover examples and patterns of different documents. Along these lines, content mining has turned into a major topic in data mining.

Information retrieval (IR) and Database frameworks were created parallel from the later past. IR deals with the affiliation and recovery of information with massive datasets generally content based information. Though, database frameworks have focused on question and interpretation procedures of organized information. Both handles different sorts of data; some database framework issues are for the most part not showed in an information recovery framework, for instance, simultaneousness control, recovery, a trade organization, and update. In conventional database frameworks, we can’t locate any normal data recovery issues, as related inquiry in view of watchwords and unstructured documents. The major issue in data recovery framework is to follow a document in a complete arrangement of gathering data in view of a customer's inquiry. In this issue client has a modest bunch of decision to pull the
significant data from the put away data, this fits best if the client has a same specially appointed data require, an IR framework can likewise make a move to push a recently discovered data thing to a client in the event that it is like a customer's question. Such a data access procedure is called data sifting, and the framework utilized as a part of this procedure is called separating frameworks or recommended frameworks.

3.1.1. Text Retrieval Methods

These are categorized into two

- Document selection problem
- And ranking problem

The document determination technique is the one in which the inquiry is the indicating requirement for selecting the important document. The Boolean recovery technique is one such strategy in which an arrangement of the fundamental words speaks to a document and a Boolean of expressions gave by the client like tea or espresso and auto and shop. All it takes is the Boolean question and return document for this framework to fulfill the Boolean expression. Endorsing client data with a Boolean question is intense, and this recovery strategy functions admirably just when the document accumulation is surely understood by the client, and a decent inquiry could be made out of it.

The document positioning strategy taking into account the request of pertinence is utilized to rank the documents. These are more important than the determination techniques for there are all the more best in class recuperation structures present, which give a rank summary of the documents. Polynomial math, rationale, likelihood and measurements are different numerical instruments on which numerous positioning techniques created. The level of significance of a document with the score in view of the data like a recurrence of words in the document and to inexact it is the primary objective. To quantify the level of pertinence, it is intense.

3.1.2. Cluster Analysis

The way toward gathering an arrangement of physical or abstract objects into classes of comparable objects is called clustering. A cluster is a gathering of data objects that are like each other in the same cluster and are not at all like the topic in different clusters. Classification assumes a noteworthy part in separating gatherings or classes of objects. This is finished with the earlier learning of preparing tuples or huge data sets or various types of examples, in which classifier can show every gathering. It is regularly more alluring to continue in the converse bearing: first segment the arrangement of data into gatherings in view of data closeness (e.g., utilizing clustering), and after that appoint names to the generally little number of gatherings. Additional purposes of enthusiasm of such a clustering based system are that it is flexible to changes and safeguard single out accommodating features that perceive various gatherings. Via is computerizing clustering, we can recognize thick and inadequate districts in object space and,
hence, find general dissemination designs and an intriguing relationship between size's data qualities.

In view of applications, clustering can likewise be characterized as data segmentation as it can classify substantial data sets into gatherings as indicated by their comparative features. Clustering can likewise be utilized for discovering data far from limits or cluster edge values as this may likewise be a state of enthusiasm from customer's inquiry. Starting now Data clustering is under fast change. A cluster examination instrument in light of k-mean, k-medoids and a few different strategies have additionally been incorporated with numerous measurable investigation programming bundles or frameworks, for example, S-Plus, SPSS, and SAS. In machine learning, clustering is a case of unsupervised learning. As it tries to gain from perceptions rather than cases [13]. More endeavors were seen in discovering best, viable and proficient cluster examination techniques in the territory of data digging for immense databases. Adaptability of clustering was a standout amongst the most noticeable components to achieve.

Here describing the data mining these are the typical requirements of clustering in data mining.

**Dealing with different types of attributes:** Most algorithms are designed to cluster numeric data. Applications may require clustering the other data types, such as binary, nominal and the original data or the mix of all these data types. As of late clustering systems are utilized for the complex information sorts, for example, charts, groupings, pictures, and documents.

**Discovery of clusters with arbitrary shape:** Cluster based algorithms can be defined based on Euclidean or Manhattan distance measures. The algorithms in view of such separation measures tend to discover circular clusters with comparative size and thickness. The cluster might be of any shape.

**Ability to deal with noisy data:** Most of the datasets contains outliers, missing, unknown, or erroneous data. If some reading may be erroneous due to interfaces from the surrounding transient objects. The clustering algorithms can be delicate to such clamor and may create low-quality clusters. So, robust to noise clustering methods are needed.

**The capability of clustering high-dimensionality of data:** A dataset contains numerous dimensions or attributes. In the clustering documents like each watchword can be viewed as a measurement, and there are regularly a large number of catchphrases. Most clustering algorithms are extraordinary at dealing with low-dimensional data, for example, datasets including just a few measurements. So, finding the clusters of data objects in a high dimensional space is a challenging, particularly considering that such data can be extremely meager and highly skewed.

**Constraint-based clustering:** In true, applications may need to perform clustering under different sorts of requirements. Accept that your occupation is to pick the ranges for a given number of new automatic teller machines (ATMs) in a city. To settle on this, you may cluster families while considering prerequisites, for instance, the city's streams and turnpike...
frameworks and the sorts and number of customers per cluster. A testing undertaking is to discover data bunches with great clustering conduct that fulfill indicated imperatives. The following are some of the aspects with which the clustering methods can be compared.

**Separation of clusters:** A few techniques parcel data objects into fundamentally unrelated clusters. At the point when clustering clients into gatherings so that every gathering is dealt with by one administrator, every client may have a place with stand out gathering. In some different circumstances, the clusters may not be select, that is, a data item may have a place with more than one cluster. For instance, when clustering archives into themes, a report might be identified with different points. Therefore, the points as clusters may not be restrictive.

**Similarity measure:** A few strategies decide the likeness between two articles by the separation between them. Such a separation can be characterized by Euclidean space. In different techniques, the comparability might be characterized by availability in light of thickness or contiguity, and may not depend on the total separation between two articles. Comparability measures assume a central part in the configuration of clustering strategies. While separation based strategies can frequently exploit enhancement systems, thickness and congruity based techniques can regularly discover clusters of self-assertive shape.

**Clustering space:** Numerous clustering strategies look for clusters inside the whole given data space. These techniques are helpful for low-dimensionality data sets. With high dimensional data, be that as it may, there can be numerous unessential traits, which can make closeness estimations inconsistent. Therefore, clusters found in the full space are regularly useless. It's regularly better to rather look for clusters inside changed subspaces of the same data set. Subspace clustering finds clusters and subspaces (frequently of low dimensionality) that show object similitude.

### 3.1.2.1. Major Clustering Methods

There exist many clustering algorithms in the literature [14]. Classifying the clustering strategies as the techniques may cover so that the features from a few classifications are gotten. The classifications of real clustering strategies are classified into:

**Partitioning Methods:**
In this method in the event that we consider n as objects of information tuples and K as the quantity of segment, then \( k \leq n \) speaks to the cluster shaped, this framed cluster ought to satisfy underneath prerequisites:

- a) Each group essential have at least one object d.
- b) Each object must belong to exactly one group.

Partitioning begins with a strategy called partitioning technique which can be enhanced by iterative movement systems. This procedure conveys object in the same cluster, which are up
and coming to each other. In the K-mean algorithm, the cluster is characterized by the mean estimation of items. These strategies find imperative in finding round formed clusters in little to medium estimated databases.

Hierarchical Methods:
A hierarchical strategy is the one which makes hierarchical decay of the items in the given set. This strategy is ordered into

a) Agglomerative approach

The agglomerative methodology is likewise rung the base methodology, begins with every item by framing a different gathering. It blends the items near each other until all gatherings are converged into one or an end condition holds. In this technique, it is clear that a stage can't be made fixed. The charges for calculation is lesser which does not influence the combinatorial number [15]. The methodologies for enhancing the nature of clustering is: to perform an examination of item linkages at apportioning and to incorporate agglomeration and different methodologies.

b) Divisive approach

It is also called top-down approach, in which all the objects are in the similar cluster. In each succeeding iteration, a cluster is split into smaller clusters, until eventually each object is in one cluster, or a close condition holds. Hierarchical clustering techniques can be separation based or thickness and congruity based. Different augmentations of progressive techniques consider clustering in the subspaces also.

Density-Based Methods

The distance between the articles lay the premise for dividing techniques. It can just discover circular molded clusters, yet not discretionary. The general strategy is to develop clusters until thickness exists in such that the range of the area contains a minimal number of focuses so that self-assertive shape clusters could be utilized. For instance, for every information point inside a given cluster, the area of a given sweep needs to contain no less than a base number of focuses. Such a strategy can be utilized to sift through commotion or anomalies and find clusters of discretionary shape. The Density-based strategies can partition an arrangement of articles into various selective clusters or a chain of the importance of clusters. Regularly these techniques consider specific clusters just yet not the fluffy clusters. These strategies can be stretched out from full space to subspace clustering. DBSCAN and expansions are thickness based strategies developing clusters for a thickness based examination.

Grid-Based Methods

These quantize the thing space into a steady number of cells which outline a lattice structure, and all the clustering counts are done on it. Quickest handling time and reliance just
on cells in the quantized space are the favorable circumstances. STING is a normal network based technique. Wave Cluster applies wavelet change for clustering examination and is both lattice based and thickness based. This methodology frequently a productive approach an excessive number of spatial information mining issues, including clustering. Hence, network-based strategies can be coordinated with other clustering techniques, for example, thickness based techniques, and the progressive techniques.

**Model-Based Methods**

Model-based strategies will conjecture a model for each one of the clusters and will locate the best attack of the information to the given model. A model-based algorithm may discover the area of a cluster by developing a thickness capacity which mirrors the spatial dissemination of the information focuses. It additionally brings about a method for deciding the quantity of clusters automatically on the premise of standard measurements, considering "commotion" or anomalies. Thus it yields in strong clustering strategies, depending on thickness finding strategy and stepwidth for not sending clusters.

A couple of the clustering algorithms consolidate the thoughts of various clustering strategies, as an aftereffect of which the characterization of a given algorithm as remarkably having a place with one and only clustering technique classification gets to be troublesome can also effort for the core components in general view and could not have effected. Then again, there are a couple of uses which may have clustering criteria which require the combination of a few clustering strategies. Notwithstanding the aforementioned classifications of clustering strategies, there are two more classes of clustering undertakings which need uncommon consideration.

**Clustering High-Dimensional Data**

This is especially an undertaking of the most noteworthy significance in cluster investigation as a few applications require the examination of articles which have countless or measurements. Numerous measurements may not be important. As the quantity of measurements builds, the information turn out to be progressively inadequate, so that the separation measured between sets of focuses turns into a push to register and the normal thickness of focuses anyplace in the information is prone to be low. In this manner a prerequisite for working up the other clustering framework for high-dimensional data exists. Inner circle and PROCLUS are two persuasive subspace clustering strategies, which hunt down clusters in subspaces (or subsets of measurements) of the information, as opposed to over the whole information space. Frequency or the redetection of the clusters are the example based clustering, another clustering system which separates successive unmistakable examples among subsets of measurements which happen regularly. It makes utilization of such examples of collection articles and for producing significant clusters. PCluster is a case of continuous example based clustering that gatherings objects in view of their example likeness. PCluster is nothing but a pattern cluster and it is a generalization of the subspace clustering. The PCluster model can be
used for the mining of clusters of an object that exhibit coherent patterns on a set of attributes [26].

**Constraint-Based Clustering**

This is another kind of clustering methodology which performs clustering by joining the client determined or application-situated limitations. Every requirement either indicates a client's desire, or depicts "properties" of the craved clustering results, and accordingly giving a viable intends to speaking with the clustering procedure for example if $x>10$, and the property as a constraint or the functionality may be at the end of the constraint process, or further sensor from view as further input possibility or the constraint. Distinctive sorts of imperatives can be indicated, either by a client or according to the prerequisites of the application. Our exchange here will be focused on spatial clustering with the presence of hindrances and cluster under client determined imperatives. Besides, semi-supervised clustering is portrayed, which applies, for occurrence, pairwise objectives, (for instance, sets of the event set apart as having a spot with the same or assorted clusters) remembering the final objective to upgrade the way of the resulting clustering. In the accompanying segments, we analyze each of the above clustering techniques in the subtle element. We additionally present algorithms that coordinate the thoughts of a few clustering strategies.

### 3.2. Machine learning

Machine learning is by and large taken to envelop automatic registering method in light of sensible or parallel operations that take in an errand from an arrangement of cases. Here we are simply focusing on characterization. Machine learning plans to produce grouping expression sufficiently basic to be seen effortlessly by the human. They should mirror human thinking adequately to give understanding into the choice procedure. Like actual methodologies, foundation knowledge might be abused in growing, yet the operation is expected without human intercession [16]. For instance, a regular machine learning the issue is to program a PC so it can automatically perceive manually written postal codes on mail subsequent to learning from an arrangement of cases. Machine learning is a quickly developing order.

Executed cases can be certain or negative. The learning algorithm then expands on the kind of case, on the size and significance of the foundation knowledge, on the representational issues, on the assumed way of the concept to be gained, and on the fashioner's experience. A fundamental necessity is that the learning system ought to have the capacity to manage the blemishes of the information. Illustrations will frequently contain a specific measure of commotion errors in the depictions or in the characterizations.
We have three different types of machine learning.

**Supervised Learning:** In order to predict new values the framework needs to take in a mapping between the inputs and yields. In the event that in a class of marked and unlabelled examples, in one of the methodology, named examples are utilized to learn class models, and unlabelled examples are utilized for defining the limits between the classes. For a two-class issue, we can think about the arrangement of cases having a place with one class as the positive cases and those having a place with interchange class as the negative illustrations

**Unsupervised Learning:** In Unsupervised Learning, the distributed input data are extracted by the system. And it is essentially a synonym for the clustering. Ordinarily, we may utilize bunching to find classes inside the data. For instance, an unsupervised learning strategy can take, as information, an arrangement of pictures of written by hand digits. Assume that it discovers 10 groups of data. In any case, consequent to the readiness data are not named; the scholarly model can't let us know the semantic importance of the groups found [17].

**Reinforcement Learning:** The framework needs to take in a strategy such that every activity ought to bring about a maximal order.

3.2.1. **Multilayer Perceptron (MLP)**

Covers theorem on distinguishableness obviously expresses that "If an intricate example arrangement information which can't be straightly detachable in lower measurement when anticipated into higher measurement than information can be directly isolated. Given that space is not thickly populated".

A MLP made out of Input Layer, a few Hidden Layer, and Output Layer. Info information is anticipated into higher measurements by means of concealed layer, on the off chance that this shrouded layer is ably picked, then yield neuron can figure out how to group the information accurately.
The hidden neuron and yield neuron plays out a weights whole of their info called net enactment and applies the transfer function to this sum to obtain the output. The most used transfer functions are a Logistic function, Threshold function, linear function and Hyperbolic function [17].

The learning principle in MLP is by Back Propagation. As shown in the fig 23, when the output $O_k$ is not equal to the desired output $T_k$, the information are back spread through yield layer to concealed layer and weights are redressed in the shrouded layer such that information is directly classified. This back spread blunder speaks to the rate of mistake in the yield layer that is brought about by the concealed neuron. The concealed layer is by and, large used to make a bottleneck, constraining the system to make a model of the framework producing the information, with the capacity to sum up to past inconspicuous example.

The back propagation algorithm comprises of food forward pass suggested for preparing the activities in all layers. The $w_{jk}$ implies that the shrouded neuron j took an interest a great deal in the mistake made in the yield neuron k.

For the hidden neurons

$$y_j = f(net_j(x)) = f\left(\sum_{i=1}^{N_j} w_{ij} x_i + b_j\right)$$

For the output neurons

$$O_k = f(net_j(y)) = f\left(\sum_{j=1}^{N_k} w_{jk} y_j + b_k\right)$$

And of a feedback pass to propagate the error of the network
\[ \delta_k = (t_k - o_k). f'(net_k) \]

\[ \delta_j = f'(net_j). \sum_{k=1}^{N_k} w_{jk}. \delta_k \]

### 3.2.2. Radial Basic Function (RBF)

The RBF system comprises a layer of units performing straight or nonlinear elements of qualities. Its structure seems to be like MLP with one hidden layer called as feature space, and the information data of measurement \(d\) is changed through \(\phi\) into feature space. In the event that the function \(\phi\) is picked effectively the data turn out to be linearly separable. The most famous function utilized is a Gaussian function due to its localized properties as maximal for the middle, decreasing to zero for bigger distances. Each hidden unit specializes in a limited part of the input space; the resulting code is scanty (only a few \(\phi_i(x)\) are different from 0).

The output of linear classifier \(y\) is characterized as

\[ y = F(x) = \sum_{i=1}^{N} w_i . \phi_i(x) \]

The function \(\phi\) is called Radial-Basis Function, as it is contained near to its center and symmetric.

The function utilized for the hidden neuron is Gaussian fixated on every training illustration

\[ \phi_i(x) = \exp(-\|x - \|^2) \]

The classification problem can be rewritten as
\[ \phi, w = t \]

Where \( \phi = \{\phi_{ij} = \varphi(\|x_j - x_i\|)\}_{i,j=1}^{N} \) the interpolation matrix, \( w \) is the linear weight vector, and \( t \) is the desired output.

Radial Basic Function uses a cross-validation technique to handle noise. As error on the cross-validation set increases, the algorithm quits training, which is most invaluable when contrasted with different algorithms. The hidden layer is figured through a solitary function \( \varphi \), yet not a progression of weights as in MLP.

### 3.3. Support Vector Machine Module

The SVM finds the hyperplane utilizing support vectors and the edges. Despite the way that the planning time of even the speediest SVMs can be amazingly direct, they are profoundly correct, inferable from their ability to model complex nonlinear decision limits. They are significantly less inclined to overfit than other strategies. SVMs can be utilized for numeric expectation and in addition arrangement [18]. They have been connected to various zones, including written by hand digit acknowledgment, object recognition, and speaker recognizable proof, and benchmark time-arrangement prediction tests. It belongs to the broad family of kernel-based learning algorithms, and the algorithms can be used for deterioration as well as classification algorithms. SVM is like logistic regression as in it tries to separate data between groups. SVM is thought to be the best stock classifier. SVM accompanies Linear SVM by considering when two gatherings of data, a straight line will separate into two parts, example, the data is linearly separable as shown in Error! Reference source not found..

![Figure 22 Linearly separable data is shown in frame A. B, C and D show possible valid lines separating two classes of data](image)
Basically, the separable case can be defined as for example the input space \( x \in x_i \), where \( x_i \) is the set of training tuples with the associated class labels, \( y_i \). Output space \( y_i \), each \( y_i \) can take one of the two values i.e., \( y_i \in \{-1, +1\} \) in the case of classification, \( w \) is the weight containing the weight coefficients for a specific hyperplane and \( b \) is the basic term or threshold. Support vectors are the data points that stay nearest to the decision surface on the margin. The maximize Margin forms as below expressed.

![Figure 23 Maximize margin](image)

The \( x_i \) is the \( i^{th} \) preparing case, and \( y_i \) is the right yield of SVM for the \( i^{th} \) preparing illustration. Estimation of \( y_i \) is +1 for positive and -1 for negative illustration. Support vector machines are a sort of classifier. They are called machines since they create a parallel choice, they are choice machines. SVM tries to maximize margin by solving a quadratic optimization problem.

In the meaning of the edge, the briefest separation from a hyperplane to the opposite side of its edge is proportional to the most limited separation from the hyperplane to the inverse side of its edge, where the sides of the edge are comparable to the hyperplane [19]. A separating hyperplane can be written as

\[
W \cdot X + b = 0
\]

Where \( W \) is the weight vector, namely, \( W = \{w_1, w_2, \ldots, w_n\} \); we can rewrite the above Equation as follows.

\[
w_0 + w_1 x_1 + w_2 x_2 = 0
\]

The weights will be adjusted so that the hyperplane defining the sides of the margin can be given as follows

\[
H_1: w_0 + w_1 x_1 + w_2 x_2 \geq 1 \ for \ y_i = +1
\]

\[
H_2: w_0 + w_1 x_1 + w_2 x_2 \leq -1 \ for \ y_i = -1
\]
In this way, any tuple that falls on or above $H_1$ has a place with +1, and any tuple that falls on or underneath $H_2$ has a place with class - 1. Finally by combining two inequalities of the equation, we get

$$y_i(w_0 + w_1x_1 + w_2x_2) \geq 1, \quad \forall_i$$

Any training data that are present on the hyperplanes are called as support vectors.

### 3.3.1. SVM in Text Detection and Character Extraction

There are different algorithms proposed for the text detection and localization. These various algorithms for text detection and localization may roughly divide into three types. These includes

**Gradient-Based Algorithms:** This method assumes that the text image has a good contrast with the background, and its edge should be obviously clear. The connected component based algorithms consider each character as connected component. This method approaches generally extracts characters based on their color or edge information. So, many text detection methods which are based on the gradient or color segmentation are classified as a connected component also.

**Texture and Structure synthesis:** This method utilizes different textural and structural features to distinguish texts from the background. In the case of perfect feature detection, these methods show more efficiency in the compound backgrounds. To detect the text from an image, the text blocks has to be determined by extracting the characters.

**SVM Classification**

SVM algorithm can be applied to binary classification or the multi-class classification. To solve the multiclass classification problem. First, it is being translated to binary classification. Through constructing and combining several binary SVM classifications, a multi-class SVM can be realized. There exist two types of multiclass SVM, one-versus-rest, and one-versus-one. And another method is by considering only one svm. Here we optimize solving of multi-class classification parameters into an optimal question and realize the output of the multi-class classification at a time though resolve the optimal question. The disadvantage of this method is, if it considers a large data for training, then it consumes a lot of time to process the data. In the above, we have the seen the binary classification of the data. The mathematical model of SVM is as follows.

$$\min \frac{1}{2}(w, w) + C \sum_{i=1}^{l} \xi_i$$
**Text Classification in SVM:** High generalization capacity of the technique makes it especially, which is suited for high-dimensional information, for example, content. SVM outperforms most of the other classification algorithms in the text categorization.

**Classification algorithm:** In the given set of examples \((x_1, y_1), \ldots, (x_l, y_l), x \in R^N, y_l \in \{-1, +1\}\). To find the decision function \(f_w, b\) with the following properties

\[
y_i((w \cdot x) + b) \geq 1, \quad i = 1, \ldots l
\]

In many of the practical applications, a separating hyperplane does not exist. To allow for the possibilities to violate the above equation, the slack variables are being introduced such as

\[
\xi_i \geq 0, \quad i = 1, \ldots l
\]

To get

\[
y_i((w \cdot x) + b) \geq 1 - \xi_i, \quad i = 1, \ldots l
\]

The support vector approach for decreasing the generalization error comprises of the accompanying. To solve the problems, the equation is.

\[
\text{Minimize: } \varphi(w, \xi) = (w \cdot w) + C \sum_{i=1}^{l} \xi_i
\]

From the above equation, minimizing the first term tends to minimize abound on VC-dimension and to minimize the second term corresponds to the misclassification error.

To evaluate the effectiveness of the approach, a comprehensive performance has to be done. After training every stage of svm, we got corresponding support vector and built the optimal
separating hyperplane. Notwithstanding an extremely number of components, svm is utilized as a classification algorithm, and its execution is given by two parameters, C is the punishment forced on preparing information sets that are on the wrong side of the choice limit, and p is the exactness edge.

To improve the classification performance, the following method can be done with the value of C. Each data set is split into 70% and 30% for training and testing. This is done several times using a stratified sampling scheme to obtain different trails for each data set. The decision threshold, p, can be configured to control precision and recall for the different tasks. Increasing in p, results in fewer test sets meeting the criterion. And this usually increases the precision but increases the recall. The LIBSVM software area used to implement the svm classifier.

Comparing the performance of the classification methods, by assuming standard performance measures such as precision and recall, which combines both in the following way:

\[
\text{Recall} = \frac{\text{number of correct positive predictions}}{\text{number of positive examples}}
\]

\[
\text{Precision} = \frac{\text{number of correct positive predictions}}{\text{number of positive prediction}}
\]

\[
F_1 = \frac{2 \times \text{Recall} \times \text{Precision}}{(\text{Recall} + \text{Precision})}
\]

LIBSVM
A Library for support vector machines. Currently, LIBSVM is one of the most using svm software. All the svm formulations funded in LIBSVM are quadratic minimization problems. It provides special settings for an unbalanced data.

C-Support Vector Classification
In the given training vectors \( x_i \in R^n, i = 1, \ldots, l \), in two classes and an indicator vector \( y \in R^l \) such that \( y_i \in \{1, -1\} \), C-SVC solves the following primal optimization problem.

\[
\min_{w,b,\xi} \frac{1}{2} (w)^T w + C \sum_{i=1}^{l} \xi_i
\]

Subject to \( y_i (w^T \phi(x_i) + b) \geq 1 - \xi_i \)

\( \xi_i \geq 0, i = 1, \ldots, l \)

Where \( \phi(x_i) \) maps to \( x_i \) into high dimensional space and \( C > 0 \) is the regularization parameter.
\[
\begin{align*}
\min & \quad \frac{1}{2} \alpha^T Q \alpha - e^T \alpha \\
\text{Subject to} & \quad y^T \alpha = 0, \\
& \quad 0 \leq \alpha_i \leq C, \quad i = 1, \ldots, l
\end{align*}
\]

**Performance Measures**

After solving the optimization problems, the user can apply decision functions to predict labels or the target values of testing data. Let \(x_1, \ldots, x_l\) be the testing data and \(f(x_1), \ldots, f(x_l)\) be decision values or the target values for the regression predicted by LIBSVM. The prediction results are evaluated by the following measures.

**Classification**

\[
\text{Accuracy} = \frac{\text{Correctly predicted data}}{\text{total testing data}} \times 100\%
\]

**Code Organization:**

The two most important sub-routines of training and testing algorithms of LIBSVM are `svm_train` and `svm_predict`. From the figure, for the classification, `svm_train` decouples a multi-class problem to two-class challenges and calls `svm_train_one` in several times. For relapse and one-class SVM, it specifically calls `svm_train_one`. At that point, as indicated by the SVM plan, `svm_train_one` calls a comparing subroutine, for example, `solve_c_svc` for C-SVC and `solve_nu_svc` for \(\nu\)-SVC.

*Figure 25 LIBSVM’s code organization for training*
To train SVM problems, some parameters should be specified. LIBSVM provides a simple tool to check a grid of the parameters. And it obtains a cross-validation for each and every parameter. The parameters with the most astounding Cross-Validation exactness are returned. The parameter determination device expects that the RBF (Gaussian) portion is utilized despite the fact that expansions to alternate parts and support vector relapse can be effortlessly made.

3.4. Clustering

The clustering examination is to partition a given arrangement of information or articles into a cluster, which speaks to subsets or a gathering. The partition has two properties such as Homogeneity inside clusters: the data which belongs to the different cluster should be as similar as possible. Heterogeneity between the clusters: the information which has a place with various clusters, ought to be as different as could be allowed. The participation functions don't mirror the genuine data circulation in the input and the output spaces. These may not be suitable for fuzzy pattern recognition [20]. To construct enrollment capacities from the accessible proof, a clustering method might be utilized to segment the information, and after that produce participation capacity from the resultant cluster. So that the characters with comparative components are in one cluster. In this way, in the acknowledgment procedure, the cluster is distinguished first and after that the real cluster.

K-means Algorithm

It is a simple unsupervised learning method used for data grouping or the classification when the number of clusters is known. This strategy works for a settled arrangement of characters. Given an arrangement of introductory clusters, dole out every point to one of them, and afterward every cluster focus is being supplanted by the mean point in the separate cluster. These steps are reiterated until the union. A point is allotted to the cluster which is close in the Euclidean partition to the point. It has the advantage to implementing the process easily. In each step, the distance between each point is to be calculated, but it is little difficult in terms of large data set.

\[ d(x, y) = \sum_{i=1}^{n} |y_i - x_i| \]

A variation of k means is obtained, and it is called K-median. The algorithm is being done by the following steps.

- Select k objects as initial centers
- Assign each data to object to the closest center
- Recalculate the centers of each cluster
- Repetition of the steps 2 and 3 until the centers do not change
Hierarchical Algorithms: The hierarchical clustering of documents can be carried out either divisively or agglomerative. Divisive clustering breaks on cluster down into small data. In an agglomerative cluster, individual item similarities are used as a starting point, and the operation collects similar groups into larger groups. Utilizing these strategies, classes of comparative items are fundamentally found by making pairwise comparisons among all of the data elements. The execution time for clustering the database on a binary field utilizing the agglomerative algorithms are pretty much equivalent and the execution time increments as the extent of the database increments.

Agglomerative Algorithm: For n samples, agglomerative algorithms, it starts with n clusters and each cluster contains a single point. After that, the two clusters will combine so that the closeness between them is the nearest until the quantity of clusters gets to be one or is specifies by the user [21]. Initially, it starts with n clusters, and a single sample indicates one cluster.

Divisive Algorithms: It begins with only one cluster that contains all sample data. At that point this single cluster parts into two or more clusters that have higher uniqueness between them until the quantity of clusters turns into various examples or as indicated by the client. First start with one cluster which contains all the samples. Now calculate the diameter of each cluster with the maximal distance between samples in the cluster.

Cluster Classification

The algorithm consists of the following steps:

- Clustering venture: to cluster both the training and testing set.
- Development step: to augment the dataset with meta-features made from the clustering step.
- Classification venture: to train a classifier with the extended dataset.
The above figure provides an insight into the cluster classification approach. The marked case of the preparation set is meant with + and − signs, while the unlabelled examples of the testing set are connoted with spots. A classifier prepared with the given training illustrations will see hyperplane. Each data sets (training and testing) are clustered into two non-overlapping Clusters. Within the best case, the two teams contain the positive and negative samples of the whole data set severally. By then, comparing meta-elements are multiplied to the present element vectors, and all element vectors inside the same cluster are enlarged with a proportionate meta-highlight. The dataset is changed into another direction framework. Since feature vectors within the same cluster are increased with the same attribute-value pair, these vectors are currently nearer to at least one another resulting in an increase in the dataset’s density, illustrated in Fig. (b). Expanding the inter-cluster distance thus prompts the highest Margin hyperplane B, as appeared in Fig. (c). As it were, the classifier is tuned to the testing set, and moreover, the classification productivity is anticipated to strengthen. Intuitively, the classifier with the largest margin can provide lower expected risk, i.e. better generalization.
4. Concept

In this chapter, we discuss more the concepts which are used to implement this project. To make the evaluation process automatic with fewer errors and fast processing to reduce the time. The block diagram gives the idea about the processing of this project work.

![Concept architecture](image)

The complete architecture of the working module is described in Figure 27. The module is divided into two blocks; one is the software layer and the other hardware layer. The software layer involves converter, evaluation, and the observation module. In this layer the process of OCR detection from the image and the text matching with the templates are done in the converter module, training the predicted data in SVM with the library called LIBSVM and testing the data with the comparison of the matching rules that are organized in the cluster preference and SVM approach. For the conceptual realization, the following modules are used.

4.1. OCR Detection

In this module the input is given in the form of PDF, tiff, etc., the data in this document provided as an input is used for the further processing. Here the pdf or the scanned document is being converted into the editable text data through OCR.

OCR has the ability to identify the machine printed, scanned pdf documents characters in an automated manner is applied in numerous fields. It currently encompasses two distinct areas such as pure optical character recognition, using the optical techniques such as mirrors, lenses and the digital character recognition using scanners and computer algorithms. In fact, these OCR devices can offer good accuracy with the high speed. The images are processed with OCR software and hardware. The process involves three operations such as document analysis to
extract individual character images, recognizing these pictures based on shape and finally contextual processing either to correct the misclassifications made by the recognition algorithm. The output interface is responsible for the communication of OCR system results for the further process.

![Diagram of OCR Detection](image)

The text is being extracted from the document in the process known as document analysis. The reliable character segmentation and recognition depend on upon each original document quality and registered image quality. Processes that commit to making amendments for poor quality originals or poor quality scanning includes image enhancement, underline removal, and noise removal. Image emphasize character versus non-character discrimination. Underline evacuation eradicates printed pointers and distinctive lines which can bit characters and interface with character acknowledgment, and thusly the commotion expulsion deletes parts of the picture that aren't a part of the characters.

The two essential components of a character recognition algorithm are the feature extractor and the classifier. The element examination decides the descriptors or the list of capabilities used to depict every one of the characters. The determined components are then utilized as the contribution to the character classifier. The contextual information may be utilized in recognition. the number of word decisions for a given field will be limited by knowing the content of the opposite field The results of recognition are often post-processed to correct the recognition errors The method of the post process character recognition result is to apply a spell checker or to verify the word spelling.

Initially, to start the applicant’s data processing, the documents containing the first page which was being created by the uniassist, consists of the applicant personal details, qualification details, and the detailed application information in a pdf file. To start for the further procedure, it is being converted to the text file through the OCR. This text data is used for the further processing.
4.2. SVM Module

In this module, it offers the possibility of selecting different types of kernel functions, such as a sigmoid, RBF, linear functions and the setting of the various parameters of the functions. After setting the type of the kernel and its parameters, the support vector machine is used to classify new sets of characters. To build this module we use LibSVM library.

In the image acquisition, a scanned image of the character data is acquired as an input image. In the pre-processing stage, it involves all the operations that are performed on an acquired image to produce a perfect character image that is further used for the further processing. In the next stage of the feature extraction, It is designed to extract the features from the segmented areas from the segmented areas of the image containing the characters to be recognised, that serves to distinguish an area corresponding to a letter from an area corresponding to other letters, and the width and height of a character are recognised by finding the minimum and maximum of columns and rows. Finally, the process implemented by this module is the normalization of the results obtained are correspond to the further acceptance of the vector support module.

For testing the exactness of the framework, in the principal test situation, a picture which contains some number of letters or characters.

From the OCR detected text, the data is being trained and tested in SVM. Here for each alphabet (a to z) and the number (0 to 9) it is given an index number for the data. In the text file
of the applicant’s details, the required data from the first page is taken in the form of keywords, such as name, or name, Date of Birth, E-mail address, Berwerbung number, etc., for every keyword or the string it is given an index number. For example the keyword or the OCR string Hochschule with the index number 20, we need to detect and compare with the SVM, in the program structure, it is given as for the character ‘H’ as the id or the index 30. Now by counting the number of repetitions of the character and the similar representation in the word, these are classified by counting the ids of the characters and analyzing the accuracy of the detection and finally comparing with the OCR string and the detected string in the SVM. By generating the text file as the input to the LibSVM, the given keywords are being formatted by detecting every character in the word and mapping to the assigned index.

LIBSVM is the library for the support vector machine technique. It includes several kernels with RBF, Polynomial, and a Linear kernel. Here the required data is being trained in the form of binary classification with linear separable data.

![Figure 30 H3 separated two classes with better margin than H1 and H2](image)

The method used by SVM to perform the parting is the hyperplane. W and b are vectors normal to the hyperplane and its displacement relative to the origin. We have the decision function for input x is given by the equation.

\[ Y = W \cdot x - b \]

Formally, the SVM need the solution of an optimization issue, given a training set instance label pairs \((x_i, y_i), i = 1 \ldots m\), where

\[ x \in R \text{ and } \{1, -1\}^m \text{ } y \in \{-1, 1\}. \]

Additionally, a grid search is achieved in order to find the optimal values for both variance parameter \((\gamma)\) of the Radial Base Function kernel and the cost parameter \((C)\) of SVM.
4.3. Cluster Module

In cluster module, the input data is given from the OCR text fragments, once the training process gets started optimization of the character repetition parameter is set, and the accuracy result is being observed in the result after comparing with the clusters stored in the database. Lookup module starts once the prediction request is sent. Here the comparison is done with the distance measurement and get the closest cluster for mapping the details or the selected keywords of the candidate from the first page.

The distance measurement and the border calculations are done with different formulae by considering the length of the word. We can observe different formats of certificates from different universities. For word mapping in the text string, formulae are set for two different types of word strings like long words with multiple lines. The long string of words is decided by the empty area incidence within the word string; the empty area is given a special parameter value whereas making a vector. The developed approach is to search out the connectivity, minimum distance, and therefore the border values. The words are mapped to the characters and the number of occurrences by comparing with each and every character in the word.

4.4. Connectivity Check

The unsupervised classification called clustering or exploratory information investigation. To give an exact portrayal of surreptitiously tests produced from the same dataset. Clustering algorithms parcel data into an express scope of clusters as gatherings, subsets or classes. In addition, distinctive methodologies, for the most part, prompt diverse clusters and notwithstanding for the same algorithm, parameter recognizable proof or presentation request of information examples may influence the last result [22]. Clustering algorithms are utilized for the separation, and the unpleasant however wide in assertion casing is to sort out clustering techniques as various levelling clustering and segment cluster, supported the properties of clusters produced. The primary plan depends on the measure of the separation or closeness between every pair of successions. The vicinity based clustering algorithms, either various leveled or apportioned, can bunch successions. Since numerous consecutive information are communicated in the in order structure. On the off chance that a succession comparison is viewed as a technique for changing an offered grouping to an alternate with a progression of substitution, insertion, and erasure operations, the space between the two arrangements are regularly laid out by uprightness of the base scope of required operations. During this sense, the similarity or distance between two sequences is often reformulated as associate degree optimum alignment drawback, which inserts well within the framework of dynamic programming.

Here first the given data is the output of the OCR detection is considered, i.e., selected keywords. By measuring the cluster distance and character matching in the cluster between the predicted data is done. The similarity between the characters and the number of occurrences are taken into consideration.
For example: To classify the word “Staatsangehörigkeit” if the detected word from the OCR is sometimes can be observed false character, here by considering each character and the spaces if available in the character between the word the similarity check is being done.

From the above figure, we can identify the classification between the OCR string and the predefined keyword. In the first sequence of the word, if the character matches we can assume it as \( i = 0 \), and checks with the next upcoming character by the incrementing method. If there exist the non-similarity between the characters from the OCR string, it checks with the next following character, and if the maximum number of characters matches each other, then it is a good classifier. It also compared with the repetition of the characters in the keyword. This procedure can be done for the selected keywords from the OCR strings from the first page of the document.

### 4.5. Split rules

In the first page, we can observe some of the names of the keywords are in multiple entries such as the university name, E-mail, course name and some general keywords. This can consume a lot of space in the page. So, to reduce the space and the distance between the words the split rules can be chosen.

#### Split rules for the University
- Detecting the keyword Hochschule through the connectivity method
- Make sure for the single row such as for four entries in a single line
- Remove everything after the word Deutsch
- Ensure that not to detect the short lines
- Split the first line when the Datum occurs
Ex: Jawaharlal Nehru Technological University

Split rule for the study program
- Detecting the Keyword Fach by the connectivity method
- Sometimes this can be in 3 to 4 entries, use last entries in a line until the numeric entry limits to 4

Ex. Electronics and Communication Engineering

Split rule for the general keyword
- Look for the general keyword by the connectivity
- Use the next entry

Ex: General keywords like Semester, Name, Vorname, Bewerbung number, email, etc

4.6. Evaluation Module

In this module, we can compare the results with SVM and Cluster results. First for every split element the index is being assigned and the cluster connectivity check set. By comparing with the SVM classified data and the Cluster connectivity data, the accuracy is detected. Cluster detection is more accurate than SVM detection. For every text file in the pdf, considered as a split element.

Document indexing:

As the writings can't be specifically deciphered by a classifier or by a classifier-building algorithm. Along these lines, it is important to have an indexing technique that maps a content from the information set into a compact representation of its substance. This should be uniformly applied to training, validation, and the test documents. The index is assigned to each and every split element.

4.7. Organization of workflow first page

The first page, which is generated by the uni-assist and which contains all the details of the applicant, is being checked and converted to the text file. It is checked by the OCR algorithm. The java code is used for the split rules by using the tesseract tool and training and testing the data in svm. In java to train, the data of the first page can be done by the following process.
After the process of the training in svm, the process is being executed in the PHP website integration.

### 4.8. Organization of workflow of performance sheet

Convert pdf to image through ImageMagick. ImageMagick is a free and open-source. Software suite for displaying, changing and writing formation image and vector image files. It can read and compose more than 200 picture document designs. One of the fundamental and completely actualized elements of ImageMagick is its capacity to change over proficiently and precisely pictures between various documents groups (it utilizes the order "believer" to accomplish this). The pdf to image conversion is available at the following link. Where the process execution and convert the function of the image magic is being done. http://www.imagemagick.org/script/convert.php

Once the pdf is converted to the image file, the data is tested by using the following process.

```java
ProcessBuilder pb = new ProcessBuilder("C:\\Python39\\python.exe",
        ";-u\\"unbuffered data to get faster
        ";C:\\\Users\\shantil\\Documents\\NetBeansProjects\\TesseractTestSWM\\libsvm\\easy.py",
        ";events\\="\;train.txt");
    runProgram(pb,null,true);
```

To iterate the images in the performance sheet, the following program code is used.

To iterate the images in the performance sheet, the following program code is used.

```java
$retval = exec("C:\\Program Files (x86)\\Java\\jre1.7.0\\bin\\java.exe" -jar TesseractTestText.jar \\
        ";filename, $resarr);
```

Convert the pdf file of the performance sheet to the text image by using OpenCV. Iterate all the images of the performance sheet, and for each image by using RGB filter of high RGB values remove the background to further processing.

The inRange function is used for masking of the image based on its value or based on the pixel intensity. The following program code is used for this process function

And this output is again generated to the tesseract files for the further processing, same as the first page. To realize the performance sheet, evaluation detection of first-page proceeding can be used with the selected keywords of the course. In the case of detection of the course which is relevant to the ASE course department, the course counter is being increased depending on the passed and failed courses.

### 4.9. Online step organization

The project is developed in the IDE NetBeans 7.3 by the setup of tess4j. The complete process is done in the tesseract files by converting the tiff file to the text document through OCR converter and tesseract to detect the character by the split rules. This content parts are
being given as a contribution to the SVM detailing to foresee the information and train the right character in the given information with the chosen catchphrases. This data is further compared with cluster connectivity and mapping rules are processed. The complete method takes place from the tesseract files directory. And finally, the testing is done and is exported to the application portal to the users.

In the online process, the student or the user can log in to the application portal by selecting the option of import automatic, and can import the files. And by selecting the bearbeiter option in the same application portal, one can check the OCR files of the corresponding applicant and undergo for the further process. In this portal the applicant or the user can check the current status of the application and the user can undergo for the further process to select by checking whether the candidate is applicable or not for the admission.

Additionally, in the performance sheet, if the student changes their course, here by checking the cleared subjects from the previous course and the appeared grade and semester time, by considering all these aspects, the correspondent can decide whether they can change their course or not. This can be done manually but with the increase in applications and manual editor’s time, it's long awaited process. So, to overcome this criterion, this application process is being developed.

4.10. Hardware Module

Hardware module includes servers, PC, and Mobiles. Right now, amid this anticipate, we tend to stack this instrument on the server and computer, however, extra there might be a degree to build up this anticipate to load this application for mobiles.

4.11. Multi-Criteria Mapping Based on SVM and Clustering Methods

This was the previous thesis project, in which the detection of the course names from the exam sheet is focused. As there is an increase in the number of applications for the master’s program, to reduce the time for the processing of the applications manually, this method is being implemented [12]. The applicant process actually concerned many steps. To begin with, the candidate sends the whole documentation to uni-help; from that point, the applications are gotten by the understudy help group at the college and are then sent to the individual offices. At the individual departments, the individual applications are going to be handled by conducting an intensive study on if the applicant has needed skills in his previous studies and if they match ASE study program. To begin with, the candidate sends the whole documentation to uni-help; from that point, the applications are gotten by the understudy help group at the college and are then sent to the individual offices. There exist different parameters like work expertise, education or course certifications, and German language skills; that fetch some bonus points. These points are combined within the final state. The applicant can get the admission if he crosses the required limit points. There are some web applications for process requests during this method. For any further examination, all the information received from the student and uni-assist is kept. As the variety of applications to be processed increase, the method wants much time and number of auditors.
The application documents are being converted to the editable text through converting by OCR detection process and the tesseract. Then recognition proceeds as a two-pass process. Inside the primary pass, acknowledgment of each word, thus, is attempted. Each word that is tasteful is passed to an adaptative classified as a preparation information. At that point, the adaptative classifier gets an open door for perceiving the content drop down the page with still extra precision. Since the adaptational classifier might need to learn something useful too late to make as way close to the top of the page, a second pass goes over the page. Within the second pass, the words that weren't properly recognized are created to recognize once again, thereby lead to greater accuracy of recognition. A final section resolves fuzzy areas, and checks alternative hypothesis for the x-height to locate small cap text. The text within the region is judged by the height of a median. Due to this, filtering out the blobs that are petite then some portion of the median height, let's say, diacritic marks, punctuation, and noise is safe.

**General Classification Controller:**

This is the main module that process needed output. It'll be a part of mapping module. The fundamental work concerned incorporates making records of required arrangement, discussion of configurations to outline positions inside the database and servers. Since the records are created from the different apparatuses and software's, the last yield can utilize these documents to deliver or anticipate a few results from these records, in this manner the information configuration of this document assumes an imperative part amid this strategy.

![General Classification Segments](image)

In the mapping module, there exist two sorts of files that are generated in an application and compared with the svm and cluster classification, with an entire course list of university of a specific study path to the courses that may be mapped to ASE connected courses. In this module, LIBSVM is going to be triggered on and information training, testing, and classification. The SVM in this module is implemented from the input data from the OCR and DB prefilled coarse loader is used to train in SVM. The information document from OCR will be classified in such the easiest way that everyone contents sections that are produced will be separated into the ASE course list or to avoid list. The machine learning bundle used in this postulation is LIBSVM. It is a bundle for support vector machine strategy. LIBSVM includes
many kernels, together with RBF (radial basis function), a polynomial and a linear kernel. Here for a given training set of instance label pair \((x_i, y_i), l = 1, 2, 3, \ldots l\) where \(x_i \in \mathbb{R}^n\) and \(y_i \in \{1, -1\}\), the SVM requires the solution of the following optimization problem:

\[
\frac{1}{2} * W^T * W + C \sum_{i=1}^{l} \xi_i
\]

Subjected to \(y_i(W^T \phi(x_i) + b) \geq 1 - \xi_i\)

\(\xi_i \geq 0\)

The training vector \(x_i\) is mapped to the higher dimensional space by the function. \(C > 0\) Is the penalty parameter of the error?

In cluster module, the contribution for streamlining is given from the OCR content pieces, once the preparation procedure begins advancement of character reiteration parameter is set, exactness results resolved inside the outcome document when comparing with the clusters put away in the database. Query module begins once forecast solicitation is disseminated. Here the comparison is completed with distance measuring and obtain the nearest cluster for mapping of courses to ASE.

A distance measuring and border calculations are done with the various formulae by considering the length of the word into consideration. We do have a yield from OCR that can state long lines by considering the endorsement position. The watched arrangements of testaments vary from college to college. The organizations can be in content string took after by a number and afterward again message string to a number. This sort of configuration is considered as a long line. For word mapping in clustering, formulae are set to two unique sorts of word strings are made of long words, words with 4 and 3 characters.

**Pre-filled Course Data:**

Once courses relating to ASE study path is generated that is predicated on the cluster, these courses are loaded into an online page. These are the courses which are mapped to ASE study course. There might be any improvements that might be extra to the current online page, wherever we can add a catch to choose or alter the courses and check in the event that every one of the courses we tend to get from cluster classification are right and on the off chance that we can utilize this information for further process. If we discover any disturbed or the incorrect course, we are able to remove that exact subject from the list by choosing it. We tend to generate a matrix for these courses that are mapped to ASE study path; this can have a good scaling in conclusion whether or not to simply accept the appliance or not. There are different matrix’s like work experience, certifications in computer courses and German skills that moves the scaling factor to bit higher. An OCR detected, and ASE study mapped courses are then presented as preselected input values to the auditor for an additional process of the application.
They have implemented the work on the two universities JNTU Kakinada and Hyderabad on the course modules. They have obtained the perfect results by using the algorithms and formulae in the process code. By making slight changes in the split and merge conditions, the ASE related courses of any university can be mapped. The below are some of the obtained results from this thesis project. For the university JNTUK, a complete range of courses studied by the student are fifty-nine and in this course, there are thirty-six courses that are relating to ASE study module and twenty-three, not ASE connected courses. At the initial stage of OCR detection, of these courses are given as input in tiff format. OCR is eminent in the detection of these courses, and a few alternative text strings are read by OCR. Next, come the split and merge stage to map the courses to an ASE study path made by Tesseract. Classification of text strings obtained from an OCR is completed by SVM that says if it’s with reference to ASE or not and therefore the mapping of courses to the correct course is completed by clustering.

<table>
<thead>
<tr>
<th>JNTUK</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Manual Passed input</td>
<td>59</td>
</tr>
<tr>
<td>Manual ASE Courses</td>
<td>36</td>
</tr>
<tr>
<td>Manual Not in ASE</td>
<td>23</td>
</tr>
<tr>
<td>OCR Mapping</td>
<td>59</td>
</tr>
<tr>
<td>OCR ASE Mapping</td>
<td>36</td>
</tr>
<tr>
<td>OCR Not ASE Mapping</td>
<td>23</td>
</tr>
<tr>
<td>Tesseract split and merge</td>
<td>50</td>
</tr>
<tr>
<td>Tesseract Detected ASE Courses</td>
<td>36</td>
</tr>
<tr>
<td>Tesseract Detect not ASE</td>
<td>9</td>
</tr>
<tr>
<td>University document excludes strings</td>
<td>5</td>
</tr>
<tr>
<td>Cluster mapping</td>
<td>59</td>
</tr>
<tr>
<td>Cluster ASE</td>
<td>36</td>
</tr>
<tr>
<td>Cluster not ASE</td>
<td>23</td>
</tr>
<tr>
<td>SVM Mapping</td>
<td>59</td>
</tr>
<tr>
<td>SVM ASE</td>
<td>41</td>
</tr>
<tr>
<td>SVM not ASE</td>
<td>6</td>
</tr>
<tr>
<td>SVM noise</td>
<td>310</td>
</tr>
<tr>
<td>SVM noise assigned to ASE</td>
<td>243</td>
</tr>
<tr>
<td>Not ASE courses assigned to ASE</td>
<td>23</td>
</tr>
</tbody>
</table>

*Figure 33 Data analysis of JNTUK University*
5. Implementation and Realization

The below architecture shows the complete concept process of the project and the code implementation. The complete code is being written in Java (Tesseract) in the IDE Netbeans environment. Initially, a tiff document is given as an input to OCR. OCR is used to convert the pdf to editable text by undergoing the complete process of detection from image to text matching with the templates and by approaching the split rules to detect the complete characters in the pdf file. This output is given to the svm and cluster for the further processing.

![Architecture](image.png)

After the process of getting the output from the OCR, the text file is being given to svm and cluster approaches. In svm, the formulation is done by considering both the text file and the keywords chosen in the text file with the starting and the stopping point. In the text file, the svm is iterated by performing the index for each word.

The following is one of the examples of the applicant's first page. The input is given from the OCR output and is given to formulating with svm and the cluster.

The organizational workflow of the first page to get the output of the process is based on as follows:

```java
ProcessBuilder pb = new ProcessBuilder("C:\Python33\python.exe","-u",//unbuffered data to get faster
"C:\Users\damir\Documents\NetBeansProjects\TesseractTestSVM\libsvm\easy.py",
  event="-train.txt");
runProgram(pb,null,true);
```
The below image is the first page of the applicant which is generated by the uni-assist and is used for the further processing in my thesis by considering the required information of the student.

![Sample Image of the first page](image-url)

---

<table>
<thead>
<tr>
<th>Status</th>
<th>in Ordnung</th>
</tr>
</thead>
<tbody>
<tr>
<td>HZB Art</td>
<td>allg., HZB - Ausland</td>
</tr>
<tr>
<td>HZB wie DL</td>
<td>N</td>
</tr>
<tr>
<td>Note 1</td>
<td>0</td>
</tr>
<tr>
<td>Transf. Note</td>
<td>0</td>
</tr>
<tr>
<td>HZB Note</td>
<td>2.81088</td>
</tr>
<tr>
<td>Fachsemester</td>
<td>1</td>
</tr>
<tr>
<td>Hochschule</td>
<td>Maharshi Dayanand University, Fochak</td>
</tr>
<tr>
<td>Datum</td>
<td>30.05.2014</td>
</tr>
<tr>
<td>Studiennote</td>
<td>2.81088</td>
</tr>
<tr>
<td>Deutschkenntnisse</td>
<td>nicht benötigt</td>
</tr>
<tr>
<td>Fachbindung</td>
<td>vollständig</td>
</tr>
<tr>
<td>Zulassungsbedingung</td>
<td>Semester</td>
</tr>
<tr>
<td>Abschluss</td>
<td>Master</td>
</tr>
<tr>
<td>1. Fach</td>
<td>Autom. Softw. Engineering</td>
</tr>
</tbody>
</table>

---

**Stammdaten**

- **Bewerbernummer**: 1492791
- **Nachname**: Sharma
- **Vorname**: Rekha
- **Geburtsname**: 
- **Geburtsdatum**: 09.10.1992
- **Geburtsort**: Dehgaon, Uttar Pradesh
- **Geschlecht**: weiblich
- **Staatsangehörigkeit**: Indian
- **Zweitstudium**: N
- **zuletzt in Paket**: 744

- **c/o Namenszusatz**: A2A/294, Janak Puri, New Delhi
- **Strassenzusatz**: PLZ 110058
- **Ort**: New Delhi
- **Land**: Indien
- **Email**: rekhsrhar92@gmail.com

---

**allgemeine Bemerkungen**


---

**1. Studienwunsch**

- **HZB Datum**: 30.05.2014
- **HZB Ort**: Indien
- **Schuljahre**: 12
- **Note 2**: 0
- **Mittelnote**: 2.81088
- **Stk Note**: 0
- **Stk Ort**: Fach
- **Abschluss**: Bachelor of Technology
- **Electronics and Communication Engineering**
- **Datum**: 30.05.2014
- **Studiennote**: 2.81088

---

**2. Studienwunsch**

- **Deutschkenntnisse**: nicht benötigt
- **Fachbindung**: vollständig
- **Zulassungsbedingung**: Semester
- **Abschluss**: Master
- **1. Fach**: Autom. Softw. Engineering

---

*Figure 35 Sample Image of the first page*
This image is generated to the text file through the OCR by assuming the split rules and feature extraction from the data, and it is exported for the further processing to the svm and the cluster connectivity.

For example:

Input:
Debug:
Split by PDF Splitter Stammdaten
Bewerbernnummer 1492791 c/o Namenszusatz
Nachname Sharma Strasse A2A/294, Janak Puri,
New Delhi
Vorname Rekha Strassenzusatz

For the further process, the evaluation of the matching rules are organized in the cluster preference and the svm method are considered. To format the text file the svm module and the cluster module are being used. As expressed in above figure SVM and Cluster modules are associated with the prefilled database in which we had information about the candidate identified with the course ASE. This information is used because the training our module that is finished by using LIBSVM. This training can work effectively with SVM results. Within the cluster module, we have a tendency to implemented dimensionality distance calculation that we have a tendency to used parameters like Distance, Border, and connectivity calculations.

\[ Distance = \min \text{distance} / \text{length Border} = 0.1 + 10 \times \text{distance} \]

5.1. SVM Formulation

The svm formulation is done for both the text file and the selected keywords. In the text file, the incident code is assigned to each and every character in the text file, such as for the alphabets, numbers, and the space bar.

For the above input, the each word is split by giving an index number to each different word respectively. Such as the following

Indexing:

1: Split
2: by
3: PDF
4: Splitter
5: Stammdaten
6: Bewerbernnummer
7 : 1492791
8 : c/o
9 : Namenszusatz
10 : Nachname
The above model explains the setup of the svm of the previous example. It counts the each character repetition from the indexed text output and is mapped to the selected keywords. Such as Bewerbernummer, staatsangehört, name, etc. This method is done by counting the character repetition on the word and selecting the corresponding number and finally mapping to the keywords. The above explained is a simple example of mapping the keyword "Bewerbernummer." This process is done to the complete converted file and to map the selected keywords.

5.2. Cluster Connectivity

Clustering is done primarily by comparing the output text based on its position and comparing by the neighborhood character from the database.

Cross checking with the cluster mapping and the svm mapping. Sometimes svm shows error in prediction of data or the character, based on the accuracy it is connected to a positive data in the cluster connectivity.

1: svm false (Split)
2: svm false (by)
3: cluster: svm wrong not false error (Fach)
4: svm false (Splitter)
5: svm false (Stammdaten)
6: cluster correct (Bewerbernummer)
7: svm false (1492791)

5.3. Cluster Mapping

Finally, the data is mapped based on the above processes with the assumed database and used for the further processes.

39: Staatsangehörigkeit<->Indien
154: Studiennote<->2.81088
143: Fach<->Electronics and Communication Engineering
150: Hochschule<->Maharshi Dayanand University, Rohtak
6: Bewerbernummer<->1492791
25: Geburtsdatum<->09.10.1992
36: Geschlecht<->weiblich
41: Email<->rekhshar92@gmail.com

5.4. Evaluation of Performance Sheet

Here the pdf file is converted to image through imagemagic. To remove the background color and to detect the semester term OpenCV library is used and the further process is done as of the process of the first page.

The below screen shot is the sample image of the performance sheet, which is required to evaluate and to provide an indicator for the decision of study course change. To realize the performance sheet evaluation, the detection of first-page proceeding is used by selecting the new keywords in the study course. In the case of detection of the course, it is chosen that the relevant courses with the current course and the previous course subjects are considered to decision making. So, the course counter is being increased depending on the passed and failed subjects in the course.
The following is the performance sheet “Leistungsuebersicht” process executing convert function of image magic.

```bash
$command = "C:\Program Files (x86)\Java\jre1.7.0\bin\java.exe"
  -jar pdftotifile\leistungsuebersicht.jar "$filename2" "$id";
echo "</br>$command </br>";
$retval = exec($command, $resarr);
```

Once it is converted to the image file, the images are iterated based on the OpenCV by using the following process function. The reason for using the OpenCV is, the tesseract cannot detect the character by removing the background, and it skips some of its entries. So, here I am using the OpenCV to remove the background of an image.
Now, the `inRange` function is used for masking of the image based on the image pixel intensity with the black and the white values of the characters in the selected image. It is shown in the following process function:

```cpp
while (file_exists($filenametif)) {
    // may make id like names
    $retval = exec("OpenCVTest.exe ". $filename : ". onlyimage", $resarr);
    //foreach($resarr as $value){
    // echo utf8_encode($value). "\n";
    //}
    // unlink($filename.tif);
    // finally our detection
    $retval = exec("C:\Program Files (x86)\Java\jre1.7.0\bin\java.exe -jar TesseractTestEigen.jar behind.tif", $resarr2);
    // echo "$ont: 
";
    //print_r($resarr2);
    // unlink("behind.tif");
    $ont = $ont + 1;
    $filename = $id."". "$ont". tif";
}
```

The complete process is used for semester detection of word "winter"; one option is to check for the pixel conditions (bright, dark) of some pixels per character and by positions group for word winter of the winter semester. That is to detect the page, where the winter semester is mentioned in the following 4 examples (so far without template needed). If sometimes the pixel may shift by 1 in y position of characters between the scans. First, the character signs are detected and then distance to previous is checked and pointed. By checking each pixel in a rectangle for fitting the characters and allowing for the each character shift dx, dy it is not so necessary to put the order list together and detects the new process.

```cpp
int r = 132;
int g = 120;
int b = 122;
int rh = 255;
int gh = 259;
int bh = 255;

if (argc > 1){
    image = imread(imageName, IMREAD_COLOR);
}

inRange(image, Scalar(r, g, b), Scalar(rh, gh, bh), mask);

imwrite("behind.tif", mask);
```
In the performance sheet of the candidate, by considering the subjects of the previous course study subjects and the upcoming new ASE course subjects, in these courses by considering the similar subjects, and analysing the candidates grade in those particular module of ASE course the decision can be taken by the professor whether to accept the application or not. Here in the above figure represents the performance sheet of a particular candidate belongs to the course study of Embedded Systems and he wanted to change the course to ASE. So, first by checking the similar subjects in both courses and ASE related course and the note points of that particular subject is being checked.

Here I have considered the similar course subject between the two courses ie., from the Embedded Systems course and a module subject from Automotive Software Engineering course is Software Platforms for Automotive Systems. Checking the Id number of this subject in both courses and analysing the similarity of the number matching. Now mapping the similar subject from both the course modules and finally checking the note points achieved by the candidate.

<table>
<thead>
<tr>
<th>Course Number</th>
<th>Course Name</th>
<th>Note/Mark</th>
</tr>
</thead>
<tbody>
<tr>
<td>3080</td>
<td>Software Platforms for Automotive Systems</td>
<td>5</td>
</tr>
</tbody>
</table>

If it is in the above case, then it represents the candidate have not passed or cleared the subject in the previous course and it is assumed as ‘1’ issue entry. And in this case the the applicant is not allowed to be accepted for ASE course.

<table>
<thead>
<tr>
<th>Course Number</th>
<th>Course Name</th>
<th>Note/Mark</th>
</tr>
</thead>
<tbody>
<tr>
<td>3080</td>
<td>Software Platforms for Automotive Systems</td>
<td>2 (1-4)</td>
</tr>
</tbody>
</table>

If it is in the above case, then it represents that the candidate have passed the subject in the previous course module and it is assumed as ‘0’ issue entry. And in this case, as the applicant had passed the subject in the previous course so he is allowed to be accepted by the professor for ASE course study.

**Development Work Process**

The background process of the project is as shown in the above figure. The complete process is done in NetBeans and programmed in Java. The OCR detection and the complete setup of the process undergoes in the tesseract part in tess4j and from the files which we are testing are placed in the directory of the tesseract files. From this the program code is done by selecting particular application file and by processing by the svm formulation rules and the cluster connectivity on the selected data called as the keywords. This complete analysed data is being imported to an application portal, where the user can access the webpage and undergo
further processing to check the status or to upload the files or to take the decision on the application acceptance.

Figure 37 OCR development steps
6. Results

In this chapter we can see the obtained results from the complete thesis project and discuss more detail about the web application process. The below screenshot represents the accuracy of the debugged and run code obtained when training the data of the first page by LIBSVM with the SVM classification and clustering.

Figure 38 Training data of first page from LIBSVM
The above screen shot represents the accuracy of debugged and run code obtained by the training of the performance sheet by LIBSVM without considering the merging rule.

The above graph represents the comparison of SVM and Cluster approaches for the correctness of the word strings. Here we can observe the error in SVM approach compared to Cluster based approach.
The below screen shot is the login page for the user to login with the given ID and password and edit the details of the applicant or to decision making based on the details of the applicant.

In below screen shot we can see the new split rules obtained from the course names and the university names obtained.
This page represents the output page of new split rules, where one can edit by giving the required details from an applicant’s first page.
The below screen shot is the output of the first page with the detection of OCR of first page and obtaining the svm formation rules, cluster connectivity check and finally the comparison of svm and cluster approaches for the required keywords. Here we can observe the details of an applicant of each and every keyword which belongs to one string is obtained in a single entity. Where one can understand the critical words in the document like the university name, course name etc.
The below figure is the output from the performance sheet with an issue from ASE course. In this if the candidate clears the subject related to ASE then it is mapped to ‘0’ it represents no issues and the application is accepted for the respective course and if the candidate does not clears the required subject in the previous course then it is mapped to ‘1’ representing with an issue entry. By analysing these results based on the course subjects, one can take decision to accept the application to give the admission or not. In this I have considered the subject module as “Software Platforms for Automotive Systems” in the Automotive Software Engineering course module.
Figure 45 Output of performance sheet
The above figure represents the complete processing steps of this project which I have done. The first part we can see is the development steps where the OCR detection is done by the split rules and processing the svm formulation rules and cluster connectivity check. The files are located in the directory to undergo the development process and finally used for the testing process in the application portal. The second part represents the edition steps process, where an applicant can edit the details in the application portal or the decision making to the user based on the qualifications of the applicant.
7. Summary

In this thesis work, a pre-processing stage of application process at TU Chemnitz is developed. Previous project includes the extraction of ASE related courses from the applicant’s certificate. In this current project the first page of the applicant from the uni-assist is detected. By converting the pdf files to the text files through OCR. Usually in any case, before that, an analysis of the complete strides required in preparing was precisely the automation must be utilized to diminish the time and workforces must be finished. After converting to the text files and by undergoing the split rules in svm and cluster approaches, the number of entities are reduces for the considered keywords. So that the required string is on one entity. But in the data training and the split and merge rules I have seen more errors when compared to cluster approach.

By using the atomization process, one can save more time and can check on different parts at a time.

Additionally, I have done another process on extracting the course names in the performance sheet. So, in the respective department on the need to check the performance sheet of the previous study course and evaluate by undergoing different processes based on the subjects they had cleared in the previous courses and those which are suitable to enter the new ASE courses are evaluated.

Finally came up with an idea to use OCR for pdf to text conversion and implement SVM and Cluster based approaches, integrating the connectivity rules to classify the data and the connectivity rule.

By describing this approach, we are able to alter detect keywords in the pre-processing stage and implementing the integration steps of the applying method and to formulate the foundations of automatic detection of the applying method and the process with the chosen keywords from the main points of the applicant, the first page area unit is needed.

To implement this, I have used SVM and Cluster connectivity based approaches and finally compared both the data using checking the data structure with the connectivity check with the required keywords from the applicants data and analysing the result.

With these methods we can accurately map the selected keywords from the first page and predict and train the data through the SVM and finally comparing with the cluster connectivity for the first page result to detect the better working approaches. At some point we cannot get correct results through SVM formulation, so we need to compare with the cluster connectivity to analyse the result.

For the first page the svm and the cluster approaches are used to analyse the result. Here the svm is used for comparison with the cluster approach for the quality approach comparison. Whereas, for the performance sheet, only the cluster approach is used without considering the svm.

The performance sheet evaluation is required to provide an indicator for the decision of the study course change. For the performance sheet, to detect the course names from the previous study course and the semester, I have used the Opencv to remove the background noise of the pixel and detected the word winter from the performance sheet. As the tesseract brings much
more result strings but can-not detect the string winter. To realize the performance sheet evaluation, the detection of the first page proceeding is used by assuming the new keywords related to the study course and the subjects in the course. In this case the detection of the relevant subjects in ASE is considered. From the previous course study, the subjects which are relevant to the ASE course are analysed. Based on this result, if there are the relevant courses observed with the passed criteria then the applicant is considered for the ASE course if not, it is not considered. In case of detection of course relevant for ASE, course counter is increase depending on passed and not passed courses of the applicant data.

For the implementation part, I have used Tesseract OCR and wrote the complete code in Java, which is very flexible to write the code and can be changes based on our requirements.

Finally, extracting the course names in performance sheet. OCR detection with the pixel forms. Proving the pixel based detection from Opencv. Pixel detection algorithms for Semester characters showed better result compared to tesseract detection result. So further pixel detection rules could be tested.
Future Scope

- Extracting the course names in the performance sheet
- OCR detection with the pixel forms
- Proving the pixel based detection by OpenCV
- Further pixel detection patterns for the universities are made to find the testimony and can be for the template matching of the logo
- More crosschecking of the OCR entries are needed with the data base pre-sets.
Appendix
Appendix A
Useful links

- Web resources to install NetBeans IDE in Windows:
  https://netbeans.org/downloads/index.html

- Web resources to install tesseract in multiple IDEs:
  https://sourceforge.net/projects/tess4j/

- Tesseract installation tutorials:
  http://tess4j.sourceforge.net/tutorial/

- Web resources to install LIBSVM and tutorials:
  https://www.csie.ntu.edu.tw/~cjlin/libsvm/

- Web resources to install OpenCV in windows:
  http://docs.opencv.org/2.4/doc/tutorials/introduction/windows_install/windows_install.html#windows-installation
Appendix B
Contents of the Compact Disk (CD)

TesseractTesttext  Here, I have provided complete code and the supporting files of the code implementation. It contains sample application files of the students which are to be processed and sample performance sheet along with libsvm files.

OpenCVTesto  Here, I have provided complete code to detect pdf to image file and extract the characters which are recruited and to remove the background noise in the performance sheet.

Thesis project  In this folder I have provided complete document of my thesis project along with results.
References

[1] Line Eikvil
Optical Character Recognition, December 1993.


[4] Ivan Dervisevic

[5] Nadira Muda, Nik Kamarlah Nik Ismail, Siti Azami Abu Bakar, Jasni Mohamad Zain
Fakulti Sistem Komputer & Kejuruteraan Perisian
Optical Character Recognition By Using Template Matching, Universiti Malaysia Pahang.

[6] Ray Smith
An Overview of the Tesseract OCR Engine, Google Inc.

[7] Gary Bradski, Adrian Kaehler

[8] Sunil Ray
Understanding Support Vector Machines, October 2015.

[9] Thorsten Joachims
Transductive Inference for Text Classification using Support Vector Machines, University of Dortmund, LS VIII.

[10] Ram B. Basnet, Andrew H. Sung
Classifying Phishing Emails Using Confidence-Weighted Linear Classifiers, 2010
International Conference on Information Security and Artificial Intelligence (ISAI 2010).


[12] Abhishek Diddikadi
Multi Criteria Mapping Based on SVM and Clustering Methods, August 2015.

Cluster Analysis, Data Warehousing and Data mining.


[17] Julian Vitay AI lab, Dept. of computer science, Tu Chemnitz @bookletSVM-Machine learning.


[26] Haixun Wang Wei Wang Jiong Yang Philip S. Yu Clustering by Pattern Similarity in Large Data sets, IBM T. J. Watson Research Center