UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

PEER TO PEER METROLOGICAL DATA SHARING MODEL

A THESIS

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

Degree of

MASTER OF SCIENCE

By

BHAIRAVSINGH GHORPADE
Norman, Oklahoma
2020

PEER TO PEER METROLOGICAL DATA SHARING MODEL

A THESIS APPROVED FOR THE
SCHOOL OF INDUSTRIAL AND SYSTEMS ENGINEERING

BY THE COMMITTEE CONSISTING OF

Dr. Shivakumar Raman, Chair

Dr. Janet K. Allen

Dr. Theodore Trafalis

© Copyright by BHAIRAVSINGH GHORPADE 2020
All Rights Reserved.

Acknowledgments

I would like to thank Dr. Shivakumar Raman for his support and guidance throughout my research. Dr. Raman has always motivated me to think out of the box and helped me in shaping my thought process in the right direction. Also, I would like to thank Dr. Janet K. Allen and Dr. Theodore Trafalis for being part of my thesis committee. Their guidance and suggestions have helped me marvelously in my research and academic life. I want to thank Jimmy Cannon for sharing his expertise and always encouraging me while working in REPF.

I am deeply grateful of Sustainment Technologies for funding my thesis and supporting the thesis with their expert machinists.

I want to thank my parents and my entire family for their support. A special thanks to my friends Abhishek Gupta, Abilash Ramesh, Akhil Potdar, Aditi Singh, Tayana Ghosh, and Meera Bankar, who always supported, helped, and guided throughout the journey.

I would like to express my gratitude and thank all the collaborators of this research, Abilash Ramesh, Cesar Marquez, and Tasfiq Alam.

Contents

List of Figures

List of Tables

Abstract

Present manufacturing systems often generate enormous amounts of data, that are often forgotten or lost. A major reason for ignoring such data is the heterogeneity of data. This research focuses on the heterogeneity between the manufacturing machine's capacity parameters and part design. In manufacturing factories, the machine capacity data is available in form of machine specifications, while part data is stored in 2D or 3D-CAD models. In this thesis, a framework is proposed to provide guidelines and strategies for acquiring, pre-processing, and storing manufacturing capacity data in the form of structured table-oriented database systems. The framework also proposes the extraction, preprocessing, and storage of dimensional data of Computer-Aided Design (CAD) part models into feature-based-logical storage within XML files. Such a database storage system can improve vendor search using advanced predictive modeling. Such a system is beneficial for small-medium scale machine shops for quantifying their manufacturing capability and constraints and linking such with a prospective pool of manufacturing part's designs.

Chapter 1: Introduction

Industrial revolution 4.0 is bringing unprecedented changes to the manufacturing industry. The smart systems helping the industry in making intelligent decisions to increase productivity and quality and reduce cost and work redundancy. Smart grids are improving connectivity and transparency within organizations. As a result of the integration of such smart systems in the manufacturing industry, large amounts of data are being generated across the organizations. For instance, the design teams generate product specification and manufacturing process data, production departments generate process and capacity related data, order delivery deals with supplier's specification data, and so on. The data generated from these various sources are not in the same format or structure. The data generated across these cross-functional teams of organizations are seen in various forms such as text and numerical, and this heterogeneous data is always hard to use across other disciplines within organizations (Patil et al., 2005).

Today's dynamic global market and extensively increasing product competition are making a rush towards collaborative and concurrent manufacturing systems (Verhagen, 2015). Complex products are being manufactured in pieces around the globe. Various levels of manufacturing participants contribute to such products. As shown in figure 1 low-level individual parts are manufactured from sheet metal or bar metal and such individual parts are assembled to form the sub-assemblies and such sub-assemblies are assembled further to manufacture the useful product. As the levels in collaborative manufacturing increases, the complexity of the process tends to increase. Each level has restricted mobility in-terms of design, production rate, and cost for manufacturing because each level output is input to the next level, and changes in any design can hamper further assemblies. Hence, engineering change management plays an important and critical role in cost and time to market. Usually, the lower-level participants are small scale industries, and such industries are used to stimulate the process using their overheads and resources.

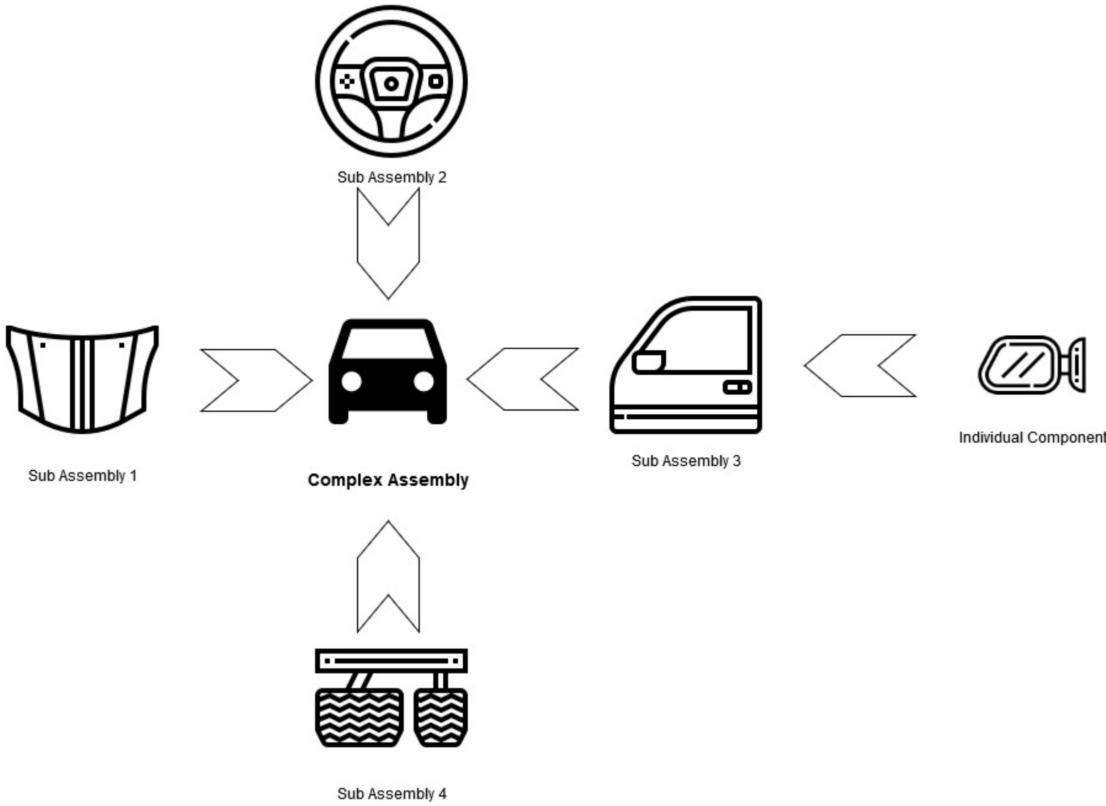

Figure 1. Levels in Collaborative Manufacturing (3 – Level System)

Figure 2 shows the process cycle in collaborative manufacturing. Assembly manufacturer issues design to the sub-assembly manufacturer. Further, this first level participant forwards the necessary part of information and design to the component manufacturer. The component manufacturer then checks feasibility and manufactures the part and supplies it to the sub-assembly manufacturer. The sub-assembly manufacturer in turn assembles the procured component and this sub-assembly is supplied to the assembly manufacturer for final assembly. In this process, significant data is generated and transferred between departments and organizations.

Due to market competition, organizations are reluctant to share their valuable data with other industries, suppliers, and sub-suppliers. But a need for a common information pool is equally important for small manufacturing businesses that are the backbone of larger organizations. Small businesses stimulate parallel manufacturing and bring down the time to market along with the cost. Small businesses are often the drivers of the manufacturing process.

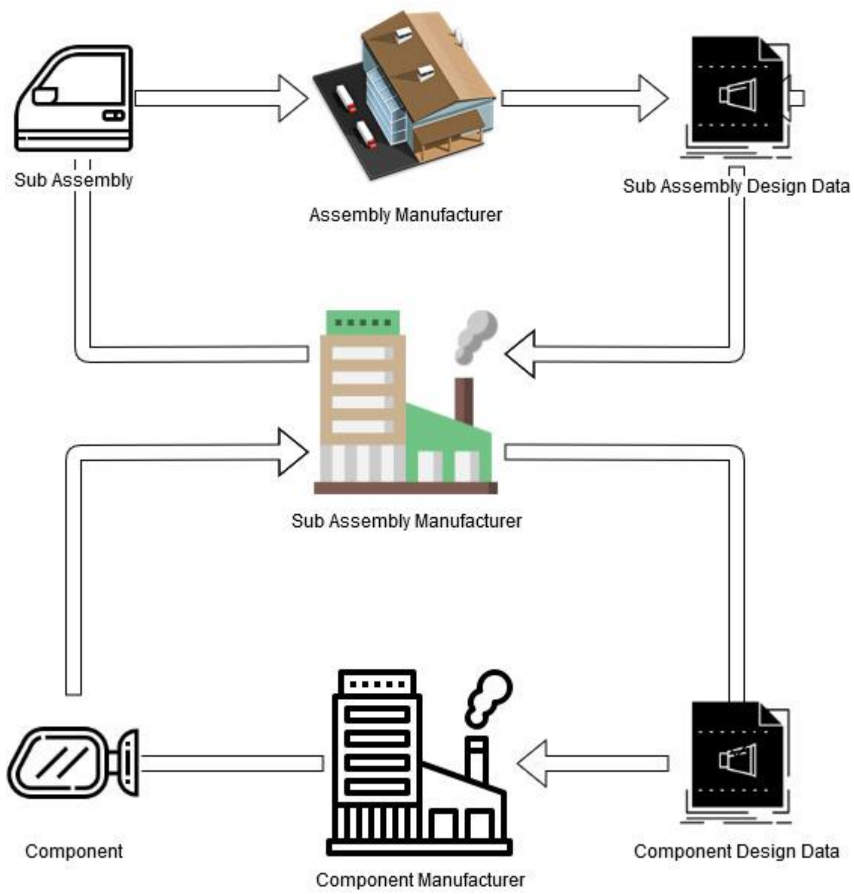

Figure 2. Process Cycle in Collaborative Manufacturing (3 – Level System)

Due to such insecurity in data sharing, the small-scale manufacturers end up having very few connections with large organizations for supplying the part. This restricts the expansion of business of small-scale industries, and on other hand, the large-scale manufactures often experience a shortage of proper capacity for manufacturing specialized or customized components for their assemblies. This in-turn costs largely to such large-manufacturers for procuring complex parts from a known and less capable supplier.

This thesis conceptualizes the framework for connecting small manufacturers using their manufacturing capability data and quantified manufacturing constraints, whenever a design of the desired part is available. The framework mainly discusses data acquisition and storage methods for manufacturing capability data of machines, and part design data.

This thesis is divided into five chapters. The first chapter discusses the introduction, the second chapter explores the background and literature survey used for building the conceptual framework. The third chapter explores the research goals and boundaries. The fourth chapter proposes the framework model, and the fifth chapter explains the conclusion and recommends the future scope for the research.

Chapter 2: Background and Literature Survey

This chapter discusses the background concepts to understand the proposed model, which includes a study of the STEP files, manufacturing machine characteristics, a brief introduction for engineering ontologies, existing part-process matching practice, and an active learning approach in training classification models. This chapter gives a summary of the literature survey for briefly understanding research work for supporting the proposed model.

2.1 Manufacturing Design Data

Design data is generated and maintained across the industries in various computer-aided design (CAD) forms. Various types of CAD systems are being used for designing the part. Throughout a product's lifecycle, CAD files store various product information such as geometrical dimensions and tolerances (Alemanni et al., 2011). The design data plays an important role during the manufacturing of the product. Design data provides insights and milestones during manufacturing (Fang et al., 2016).

As discussed in the previous chapter, the modern manufacturing industry relies heavily on concurrent manufacturing (Verhagen, 2015). This has increased the data-sharing aspects among manufacturers. CAD files are the primary source for sharing the part design between different elements of the manufacturing ecosystem. Sharing the design data entertains many challenges such as the data security and redundancy of design data. Mainly, the abundance of design data stored in the solid format often stimulates data redundancy. However, the CAD models are noncompatible for computer-aided process planning (CAPP) tools. A feature-based data format is required for using the CAD models for CAPP applications (Gao et al., 2004).

The majority of computer-aided designs exist in two formats: 2D CAD drawings and 3D CAD models. A design drawing with the design described in the two-dimensional drawing is called a 2D CAD drawing. The 2D drawings are the drawing files loaded with geometrical dimensioning and tolerancing (GD&T) information (Bijnens & Cheshire, 2019). Such 2D drawing often shows three orthographic views or a single isometric view. The more complex the parts are, the more involved the drawings are, with additional (auxiliary and sectional) views. Three-dimensional design models are used for better representation of complex features in parts. Although one shortcoming of 3D models is that they do not include the GD&T information of part. Therefore, 3-dimensional CAD designs are often accompanied by 2D design drawings. But often the design files are generated and shared differently.

CAD files are generated and shared in the following formats:(Bijnens & Cheshire, 2019)

- Generation of 2D design and sharing the 2D drawing of the product.
- Generation of 3D model of design and sharing the 2D drawing of the model.
- Generation of 3D model of design and sharing the 3D model of the product.

Although various industries use one or more of the above formats for sharing their manufacturing information, this process is still not considered efficient. Each of the formats has some disadvantages as listed below (Bijnens & Cheshire, 2019).

- Generation and sharing a 2D CAD file can be ineffective when it comes to complex assemblies. As 2D drawing gives orthographic views and this is usually hard to interpret, this can be tackled by generating other projection views. But it tends to be time consuming and inefficient.
- Generation of 3D model of design and sharing the 2D drawing of the model can overcome the above disadvantage as a small print of 3D design can be projected on the 2D drawing that can help interpretation better. It can help generate multiple different projections from the 3D model. But if there is any change in the design then the original 3D model has to be changed and the number of projections has to be generated again, and this revision should be updated, and all old versions should be discarded. Generally, in concurrent manufacturing change in design is a common factor, and changing or updating an entire set of 2D drawings every time is not so simple as these drawings are already been distributed to various participants within and outside the organizations.
- Generation and sharing 3D model design can be much more effective for change management but it has a technological disadvantage that is, many low-level participants are small-scale industries and such industries sometimes do not have the luxury of accessing the 3D model. Also, on the shop floor, the technician or machine operators are not familiar with using CAD software. Moreover, if the end-user uses a different CAD software than the designer then it is difficult to load and access the CAD file. For instance, if the 3D model design is generated in AutoCAD and the end-user uses CATIA then accessing the design file will be difficult for the end-user.

For tackling these challenges, a native CAD format developed by the International Standards Organization can be used. In the upcoming sections, STEP AP203 native CAD format is explored and used for dimensional data extraction in the proposed model.

2.1.1 Introduction to ISO STEP files

The STEP file stands for Standard for the Exchange of Product Model Data. As the name suggests the STEP format is an international standard neutral CAD format developed by the International Organization for Standardization (ISO) for geometric data exchange. The STEP file is explained in ISO 10303-21 (Al-wswasi & Ivanov, 2019)(Industrial automation systems and integration product data representation and exchange—part 203, 1994)(MALLESWARI, 2013). The STEP file extensively stores dimensional and geometric data of the part. In the STEP file, a 3D CAD model is stored in the text form using EXPRESS data modeling language (Sateesh & P, 2017). Before the STEP file, International Graphics Exchange Specification (IGES) was used as a standard CAD exchange format (Ismail et al., 2002).

The advantages of the STEP format are as follows:

- STEP is a neutral file format that can be generated from various CAD software such as AutoCAD, SolidWorks, CATIA, etc. (Industrial automation systems and integration product data representation and exchange—part 203, 1994).
- STEP format is both human and machine-readable and this can be viewed as a text file.
- The CAD model is represented in borderline representation (B-line rep.) format.
- The design surfaces are defined using specific predefined keywords (keys).

2.1.2 Structure of STEP AP203 File

The main advantage of the STEP file is, that the STEP file is machine and human-readable. To store design data in a logical order, STEP files are organized using a specific format. Understanding the structure of the STEP file is important for developing an algorithm for extracting the dimensions and feature information of the part. Moreover, STEP files are organized using various keywords, these keywords navigate the structure and part information throughout the file body. Broadly, the STEP file is organized into two separate sections (Al-wswasi & Ivanov, 2019). These sections are shown in figure 3 and explained below.

```
ISO-10303-21;
HEADER;
FILE_DESCRIPTION (( 'STEP AP214' ),
    '1' );
FILE_NAME ('cylinder_10d_50_l.STEP',
    '2020-05-20T19:21:46',
    ( '' ),
    ( '' ),
    'SwSTEP 2.0',
    'SolidWorks 2019',
    '' );
FILE_SCHEMA (( 'AUTOMOTIVE_DESIGN' ));
ENDSEC;

DATA;
#1 = ORIENTED_EDGE ( 'NONE', *, *, #126, .F. ) ;
#2 = VERTEX_POINT ( 'NONE', #113 ) ;
#3 = ORIENTED_EDGE ( 'NONE', *, *, #126, .T. ) ;
#4 = AXIS2_PLACEMENT_3D ( 'NONE', #124, #36, #102 ) ;
#5 = UNCERTAINTY_MEASURE_WITH_UNIT (LENGTH_MEASURE( 1.000000000000000082E-05 ), #47, 'distance_accuracy_value', 'NONE' );
#6 = FILL_AREA_STYLE_COLOUR ( '', #115 ) ;
#7 = FILL_AREA_STYLE ('',( #6 ) ) ;
#8 = CARTESIAN_POINT ( 'NONE',  ( 0.0000000000000000000, 50.00000000000000000, 0.0000000000000000000 ) ) ;
#9 =( NAMED_UNIT ( * ) PLANE_ANGLE_UNIT ( ) SI_UNIT ( $, .RADIAN. ) );
#10 =( GEOMETRIC_REPRESENTATION_CONTEXT ( 3 ) GLOBAL_UNCERTAINTY_ASSIGNED_CONTEXT ( ( #5 ) ) GLOBAL_UNIT_ASSIGNED_CONTEXT ( ( #47, #34, #83 ) )
REPRESENTATION_CONTEXT ( 'NONE', 'WORKASPACE' ) );
#11 = ADVANCED_FACE ( 'NONE', ( #56 ), #99, .F. ) ;
#12 = APPLICATION_PROTOCOL_DEFINITION ( 'draft international standard', 'automotive_design', 1998, #96 ) ;
#13 = DIMENSIONAL_EXPONENTS ( 1.000000000000000000, 0.0000000000000000000, 0.0000000000000000000, 0.0000000000000000000, 0.0000000000000000000,
0.0000000000000000000, 0.0000000000000000000 ) ;
#14 = EDGE_LOOP ( 'NONE', ( #97, #68 ) ) ;
#15 = VERTEX_POINT ( 'NONE', #32 ) ;
#16 = DIRECTION ( 'NONE',  ( 0.0000000000000000000, 1.000000000000000000, 0.0000000000000000000 ) ) ;
#17 = VECTOR ( 'NONE', #125, 39.37007874015748143 ) ;
#18 = DIRECTION ( 'NONE',  ( -0.0000000000000000000, -1.000000000000000000, -0.0000000000000000000 ) ) ;
#19 = DIRECTION ( 'NONE',  ( 0.0000000000000000000, 0.0000000000000000000, 1.000000000000000000 ) ) ;
#20 = DIRECTION ( 'NONE',  ( 0.0000000000000000000, 1.000000000000000000, 0.0000000000000000000 ) ) ;
#21 = ADVANCED_BREP_SHAPE_REPRESENTATION ( 'cylinder_10d_50_l', ( #146, #95 ), #54 ) ;
#22 = CIRCLE ( 'NONE', #103, 10.00000000000000000 ) ;
#23 = DIRECTION ( 'NONE',  ( 0.0000000000000000000, 0.0000000000000000000, -1.000000000000000000 ) ) ;
#24 = FILL_AREA_STYLE_COLOUR ( '', #134 ) ;
#25 = LENGTH_MEASURE_WITH_UNIT ( LENGTH_MEASURE( 0.02539999999999999897 ), #59 );
#26 = STYLED_ITEM ( 'NONE', ( #135 ), #21 ) ;
#27 = DIRECTION ( 'NONE',  ( 0.0000000000000000000, 0.0000000000000000000, 1.000000000000000000 ) ) ;
#28 = PRODUCT_RELATED_PRODUCT_CATEGORY ( 'part', '', ( #143 ) ) ;
#29 = CIRCLE ( 'NONE', #148, 10.00000000000000000 ) ;
```


Figure 3. Sections of A STEP File

i. ***Header Section:*** The header section is located at the top (starting) of the STEP file. This section includes meta-information of part and design file such as the type of STEP file, filename, software used to create the drawing, time, and date of file creation, etc. The section begins with the keyword **HEADER** and ends with **ENDSEC**.

ii. ***Data Section:*** Data section is located under the header section. As shown in figure 3, the Data section starts with the keyword **DATA** and ends with **ENDSEC**. This section stores part dimensional and geometrical information such as face numbers, edge numbers, coordinate points, and vertex points of each edge. The information in the data section is represented in a line-wise manner. Each line discloses certain information often called entity, and the type of information is identified using keywords present in each line called Entity Type. Each line begins with a unique integer. This unique integer acts as the line number of the entity in the DATA section. The line number begins with integer 1 and has '#' as a prefix. The information representation format is given as follows.

LINE_NO = ENTITY_TYPE (ENTITY)

The STEP file format stores part design data in a tree structure. The tree structure contains several predefined entity types, and each entity type has a predefined keyword called entity type. These entities and their predefined entity types are explained in the next section. Most types of entities store line numbers of the next consecutive entity type. Some entity types store dimensional information such as coordinate points or geometrical information of edge, these entities are nodes of a given STEP tree.

The entities of STEP files hierarchically store information. Each entity guides to the next lower entity and so on until the node of the hierarchical tree is reached. These nodes store various information, but information of principal interest are coordinate point locations. The part design information is organized randomly throughout the data section. However, the STEP file possesses an underlying structure that leads to extract the geometrical features. This extraction is possible by traversing the tree structure, which is explained in the following section (Ghorpade et al., 2020) (Al-wswasi & Ivanov, 2019).

2.1.3 Data Layout in STEP AP203 file

The tree structure of the STEP file is organized using keywords. These keywords describe the nature of data present in the line in the STEP file. While describing part design in a STEP file, each of these keywords navigates the information of the next hierarchical keyword using the line numbers. These keywords are defined below (Sateesh & P, 2017) (Al-wswasi & Ivanov, 2019).

i. ***CLOSED_SHELL:*** Closed shell can be considered as a gateway for part design in a STEP file. This entity defines a component as a single closed shell. Each closed model in CAD space in the given STEP file is represented by one CLOSED_SHELL. All feature's faces bind the CLOSED_SHELL together. Therefore, the CLOSED_SHELL incorporates details of all faces, edges, vertex points, and respective coordinate point locations. Further, CLOSED_SHELL divides the part into faces of the given part model. Each face gets one branch, and it expands further until the vertex point level of each of its edges. The line containing CLOSED_SHELL as keyword discloses the line numbers of all the ADVANCE_FACEs which bounds the part together.

ii. ***ADVANCED_FACE:*** The advanced face is a generic face of a given part. The number of advance faces depends on the number of faces that encloses the given part. The line with the keyword ADVANCE_FACE contains further line numbers of FACE_OUTER_BOUND/FACE_BOUND. Each Advance face contains one face outer bound. Hence, only one line number of respective face outer bound is given in this line. Moreover, this line also gives the surface type information such as "CYLINDER" for circular or curved face, "PLANE" for planar surfaces, "CONICAL" for a conical type of surfaces, and "TOROIDAL" for toroidal surfaces.

iii. ***FACE_BOUND/FACE_OUTER_BOUND:*** Face outer bound, or face bound both discloses the information of edges that develops the respective face. Each face is formed with a loop of edges. The line with the keyword FACE_BOUND navigates to the EDGE_LOOP lines for further travel.

iv. ***EDGE_LOOP:*** Each edge loop is the set of edges that forms the respective face. The line with the keyword EDGE_LOOP provides the line numbers for all the edges which disclose the represented surface.

v. ***ORIENTED_EDGE:*** Each edge in a given face is represented by one oriented edge. The line with the keyword ORIENTED_EDGE directs to the line which contains EDGE_CURVE. These edges are shown in a line with the keyword "ORIENTED_EDGE".

vi. ***EDGE_CURVE:*** EDGE_CURVE gives detailed information on ORIENTED_EDGE. Information such as vertex points and the type of edge curve is given. The type of EDGE_CURVE provides information about the type of the edge such as "CIRCLE" for curves, or "PLANE" for straight edges. This line can be representing different faces. The faces which share edges can be tracked down using EDGE_CURVEs. All entities from and below EDGE_CURVE in the STEP hierarchy are repeated for multiple faces that share the edges.

vii. ***VERTEX_POINT:*** Each VERTEX_POINT represents one vertex of the edge. Further, the VERTEX_EDGE navigates to the entity type CARTESIAN_POINT in the STEP file.

viii. ***CARTESIAN_POINT:*** The line with the keyword CARTESIAN_POINT is the lowest level entity in the hierarchical structure of the STEP file. These points define the position of each vertex point of edges in the three-dimensional space CAD space. The relative locations of vertexes of the same edge give the distance between the vertexes.

As discussed above, the keywords are standard and have a fixed definition.

Using the above-mentioned keywords, a part's design can be organized in a hierarchal structure as shown in figure 4 (Sateesh & P, 2017) (Al-wswasi & Ivanov, 2019).

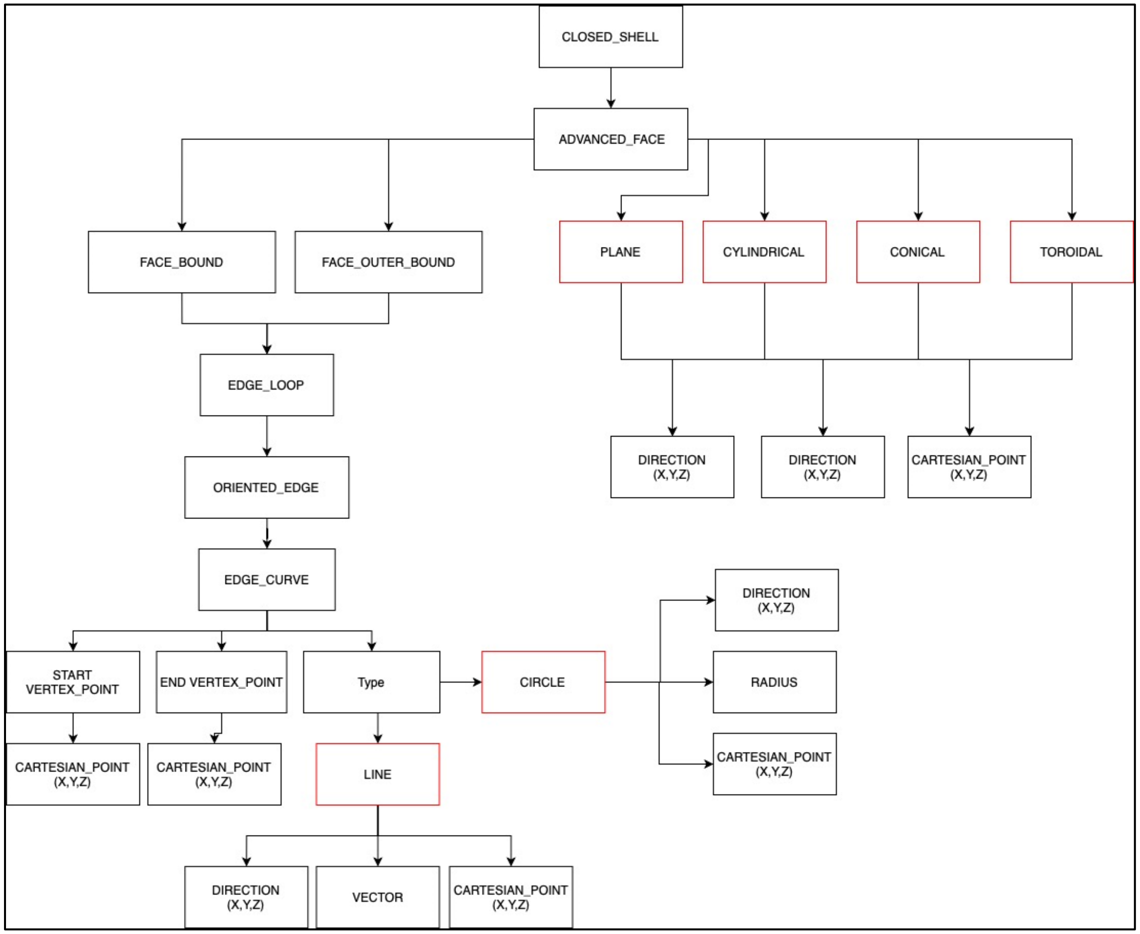

Figure 4. STEP 203 Tree Structure (Al-wswasi & Ivanov, 2019)

For understanding the structure of the STEP file, an example cylinder design with radius of 10 inches, and a length of 50 inches is considered for demonstration. The design of the cylinder is shown in figure 5 in the form of an engineering drawing. The example STEP file is exported from the SolidWorks CAD software system.

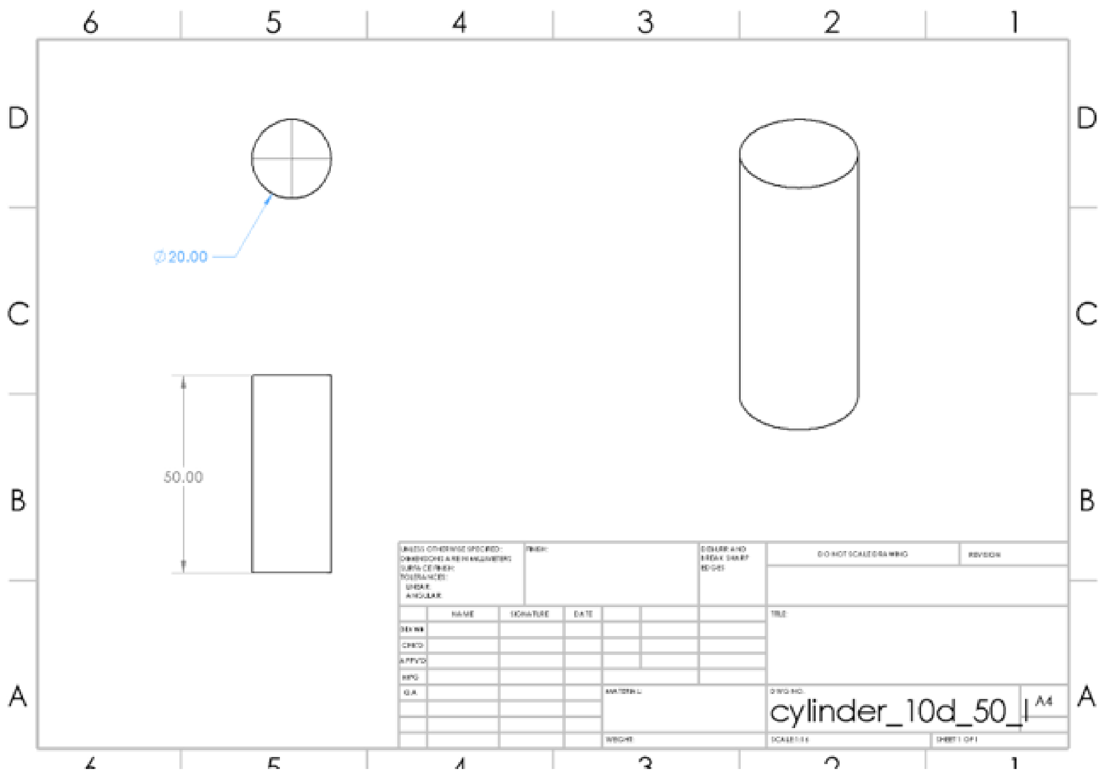

Figure 5. Different Views of Example Cylinder Part

The generated STEP file is shown in figure 7. STEP file stores design information in a scattered way. The snap shows data only to line number 38. From figure 4, the structural representation of part starts from a CLOSED_SHELL. Supposedly this closed line should come at the top of the data section in the STEP file. But the closed-shell is situated at line number 149. Therefore, to extract the exact design of a given part the tree structure needs to be followed. After carefully traversing through the STEP tree structure, a sorted STEP file looks as shown in figure 8. The advanced face #85 shown in figure 6 represents the semi-cylindrical face of the cylinder as shown in figure 9.

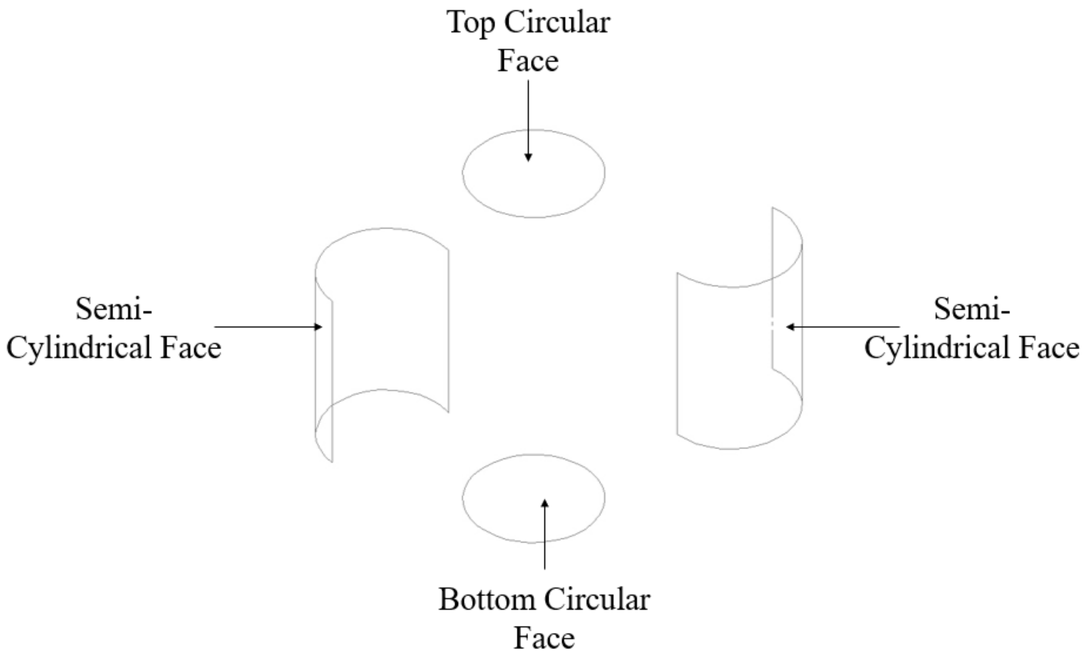

Figure 6. Cylinder Faces Represented in STEP File

```
ISO-10303-21;
HEADER;
FILE_DESCRIPTION (( 'STEP AP214' ),
    '1' );
FILE_NAME ('cylinder_10d_50_l.STEP',
    '2020-05-20T19:21:46',
    ( '' ),
    ( '' ),
    'SwSTEP 2.0',
    'SolidWorks 2019',
    '' );
FILE_SCHEMA (( 'AUTOMOTIVE_DESIGN' ));
ENDSEC;

DATA;
#1 = ORIENTED_EDGE ( 'NONE', *, *, #126, .F. ) ;
#2 = VERTEX_POINT ( 'NONE', #113 ) ;
#3 = ORIENTED_EDGE ( 'NONE', *, *, #126, .T. ) ;
#4 = AXIS2_PLACEMENT_3D ( 'NONE', #124, #36, #102 ) ;
#5 = UNCERTAINTY_MEASURE_WITH_UNIT (LENGTH_MEASURE( 1.000000000000000082E-05 ), #47, 'distance_accuracy_value', 'NONE');
#6 = FILL_AREA_STYLE_COLOUR ( '', #115 ) ;
#7 = FILL_AREA_STYLE ('',( #6 ) ) ;
#8 = CARTESIAN_POINT ( 'NONE',  ( 0.000000000000000000, 50.00000000000000000, 0.000000000000000000 ) ) ;
#9 =( NAMED_UNIT ( * ) PLANE_ANGLE_UNIT ( ) SI_UNIT ( $, .RADIAN. ) );
#10 =( GEOMETRIC_REPRESENTATION_CONTEXT ( 3 ) GLOBAL_UNCERTAINTY_ASSIGNED_CONTEXT ( ( #5 ) ) GLOBAL_UNIT_ASSIGNED_CONTEXT ( ( #47, #34, #83 ) )
REPRESENTATION_CONTEXT ( 'NONE', 'WORKASPACE' ) );
#11 = ADVANCED_FACE ( 'NONE', ( #56 ), #99, .F. ) ;
#12 = APPLICATION_PROTOCOL_DEFINITION ( 'draft international standard', 'automotive_design', 1998, #96 ) ;
#13 = DIMENSIONAL_EXPONENTS ( 1.000000000000000000, 0.000000000000000000, 0.000000000000000000, 0.000000000000000000, 0.000000000000000000,
0.000000000000000000, 0.000000000000000000 ) ;
#14 = EDGE_LOOP ( 'NONE', ( #97, #68 ) ) ;
#15 = VERTEX_POINT ( 'NONE', #32 ) ;
#16 = DIRECTION ( 'NONE',  ( 0.000000000000000000, 1.000000000000000000, 0.000000000000000000 ) ) ;
#17 = VECTOR ( 'NONE', #125, 39.37007874015748143 ) ;
#18 = DIRECTION ( 'NONE',  ( -0.000000000000000000, -1.000000000000000000, -0.000000000000000000 ) ) ;
#19 = DIRECTION ( 'NONE',  ( 0.000000000000000000, 0.000000000000000000, 1.000000000000000000 ) ) ;
#20 = DIRECTION ( 'NONE',  ( 0.000000000000000000, 1.000000000000000000, 0.000000000000000000 ) ) ;
#21 = ADVANCED_BREP_SHAPE_REPRESENTATION ( 'cylinder_10d_50_l', ( #146, #95 ), #54 ) ;
#22 = CIRCLE ( 'NONE', #103, 10.00000000000000000 ) ;
#23 = DIRECTION ( 'NONE',  ( 0.000000000000000000, 0.000000000000000000, -1.000000000000000000 ) ) ;
#24 = FILL_AREA_STYLE_COLOUR ( '', #134 ) ;
#25 = LENGTH_MEASURE_WITH_UNIT ( LENGTH_MEASURE( 0.02539999999999999897 ), #59 );
#26 = STYLED_ITEM ( 'NONE', ( #135 ), #21 ) ;
#27 = DIRECTION ( 'NONE',  ( 0.000000000000000000, 0.000000000000000000, 1.000000000000000000 ) ) ;
#28 = PRODUCT_RELATED_PRODUCT_CATEGORY ( 'part', '', ( #143 ) ) ;
#29 = CIRCLE ( 'NONE', #148, 10.00000000000000000 ) ;
```

Figure 7. Unsorted STEP File for Cylinder Part

```
#149 = CLOSED_SHELL ( 'NONE', ( #145, #63, #75, #11 ) ) ;

#145 = ADVANCED_FACE ( 'NONE', ( #85 ), #142, .T. ) ;

#85 = FACE_OUTER_BOUND ( 'NONE', #123, .T. ) ;
#123 = EDGE_LOOP ( 'NONE', ( #114, #3, #42, #106 ) ) ;

#114 = ORIENTED_EDGE ( 'NONE', *, *, #137, .F. ) ;
#137 = EDGE_CURVE ( 'NONE', #41, #2, #22, .T. ) ;
#41 = VERTEX_POINT ( 'NONE', #116 ) ;
#116 = CARTESIAN_POINT ( 'NONE',  ( 1.224646799147352862E-15, 50.00000000000000000, 10.00000000000000000 ) ) ;
#2 = VERTEX_POINT ( 'NONE', #113 ) ;
#113 = CARTESIAN_POINT ( 'NONE',  ( 0.000000000000000000, 50.00000000000000000, -10.00000000000000000 ) ) ;
#22 = CIRCLE ( 'NONE', #103, 10.00000000000000000 ) ;
#103 = AXIS2_PLACEMENT_3D ( 'NONE', #8, #57, #19 ) ;
#8 = CARTESIAN_POINT ( 'NONE',  ( 0.000000000000000000, 50.00000000000000000, 0.000000000000000000 ) ) ;
#57 = DIRECTION ( 'NONE',  ( 0.000000000000000000, 1.000000000000000000, 0.000000000000000000 ) ) ;
#19 = DIRECTION ( 'NONE',  ( 0.000000000000000000, 0.000000000000000000, 1.000000000000000000 ) ) ;

#3 = ORIENTED_EDGE ( 'NONE', *, *, #126, .T. ) ;
#126 = EDGE_CURVE ( 'NONE', #41, #107, #58, .T. ) ;
#41 = VERTEX_POINT ( 'NONE', #116 ) ;
#116 = CARTESIAN_POINT ( 'NONE',  ( 1.224646799147352862E-15, 50.00000000000000000, 10.00000000000000000 ) ) ;
#107 = VERTEX_POINT ( 'NONE', #118 ) ;
#118 = CARTESIAN_POINT ( 'NONE',  ( 1.224646799147352862E-15, 0.000000000000000000, 10.00000000000000000 ) ) ;
#58 = LINE ( 'NONE', #129, #17 ) ;
#129 = CARTESIAN_POINT ( 'NONE',  ( 1.224646799147352862E-15, 50.00000000000000000, 10.00000000000000000 ) ) ;
#17 = VECTOR ( 'NONE', #125, 39.37007874015748143 ) ;
#125 = DIRECTION ( 'NONE',  ( -0.000000000000000000, -1.000000000000000000, -0.000000000000000000 ) ) ;

#42 = ORIENTED_EDGE ( 'NONE', *, *, #43, .T. ) ;
#43 = EDGE_CURVE ( 'NONE', #107, #15, #29, .T. ) ;
#107 = VERTEX_POINT ( 'NONE', #118 ) ;
#118 = CARTESIAN_POINT ( 'NONE',  ( 1.224646799147352862E-15, 0.000000000000000000, 10.00000000000000000 ) ) ;
#15 = VERTEX_POINT ( 'NONE', #32 ) ;
#32 = CARTESIAN_POINT ( 'NONE',  ( 0.000000000000000000, 0.000000000000000000, -10.00000000000000000 ) ) ;
#29 = CIRCLE ( 'NONE', #148, 10.00000000000000000 ) ;
#148 = AXIS2_PLACEMENT_3D ( 'NONE', #144, #20, #90 ) ;
#144 = CARTESIAN_POINT ( 'NONE',  ( 0.000000000000000000, 0.000000000000000000, 0.000000000000000000 ) ) ;
#20 = DIRECTION ( 'NONE',  ( 0.000000000000000000, 1.000000000000000000, 0.000000000000000000 ) ) ;
#90 = DIRECTION ( 'NONE',  ( 0.000000000000000000, 0.000000000000000000, 1.000000000000000000 ) ) ;
```

Figure 8. Sorted STEP File of Cylinder Part

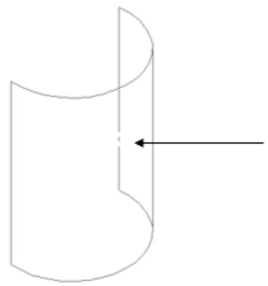

Figure 9. Semi-Cylindrical Face #85

The example cylinder's semi-cylindrical face STEP tree structure is shown in figure 10. This cylindrical part is enclosed by the CLOSED_SHELL given in line number #149 in the sorted STEP file as shown in figure 8. Within this CLOSED_SHELL, there are four advanced faces #145, #63, #75, and #11. Each of these advanced faces describes the faces of the cylinder. For instance, the tree structure of this STEP file shown in figure 10 represents the semi-cylindrical ADVANCED_FACE #145. This face and its co-ordinate points are shown in figure 11. Within this ADVANCED_FACE, there are four edges #114, #3, #42, #6. Edge #3 represents the vertical edge of this face and edge #3 represents the semi-circular top edge of the same face. These edges are shown in figure 11. Each of these faces further contains their respective vertex points and their respective cartesian coordinates. For example, edge #3 contain two vertex points #41 and #107 with coordinate location (0, 50, 10) and (0, 0, 10) respectively. These three-dimensional points describe the distance between the points i.e 50 inches. Similarly, for edge #114, vertex points #41 and #2 are shown. Here, these two edges share the same vertex point (#41). Moreover, two ADVANCED_FACEs can share the same edge if there are adjacent to each other. This gives the orientation information of edges and further orientation information of faces.

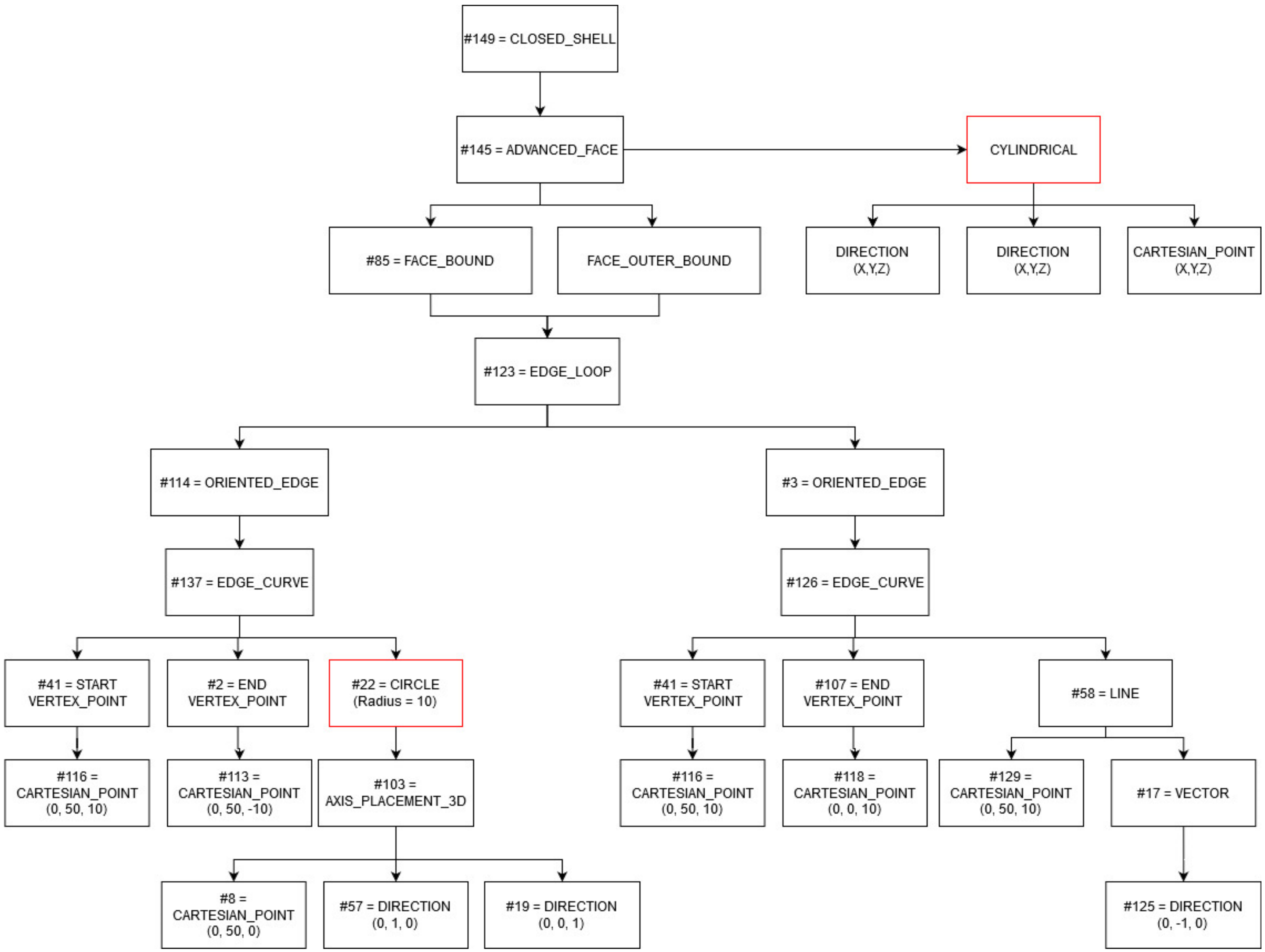

Figure 10. STEP Tree Branch for Semi-Cylindrical Face of Cylinder Part

On the other hand, Edge #114 is a curve, which is represented under the type of edge in line #22. This entity gives the radius of the curved edge (10 inches). Hence, for such curved edge's radiuses can be extracted using this entity.

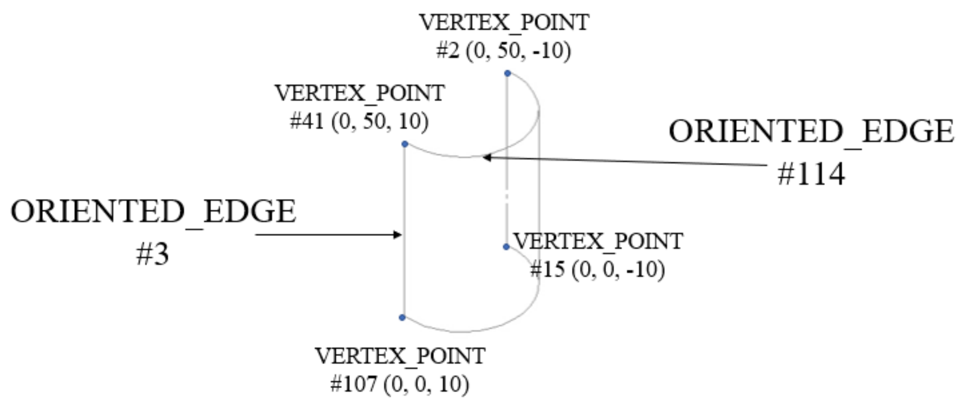

Figure 11. Semi-Cylindrical Advanced Face with Cartesian Coordinates

Each face in the part is represented and can be tracked as illustrated above. Systematically traversing the STEP tree and storing appropriate data is extremely necessary for extracting dimensions of the given part. A face-based logical XML storage is used for storing the part which represents the structure and respective faces in the hierarchical model.

2.2 Scope of XML Files

Markup languages are being used rather extensively in recent years. These are systematic files for storing the content from electronic documents so that the machine can read and process it for further applications. The definite and retrievable content-based structure makes markup language a good fit for integrating with system and human interaction atmosphere (Schools of advanced study university of London, 2020).

Various types of markup languages are being used in different systems and applications. Some popular markup languages are listed below (Schools of advanced study university of London, 2020).

- HTML (HyperText Markup Language)
- XHTML (Extensible HyperText Markup Language)
- MathML (Mathematical Markup Language)
- KML (Key whole Markup Language)
- SGML (Standard Generalized Markup Language)

After the preliminary study, this research report focuses on the XML markup language. XML is used in the industries for various applications due to its compatibility and ability to boost the effectiveness of search engines as compared to other file storages (Gil & Ratnakar, 2002) (Abiteboul et al., 2014).

2.2.1 Advantages of XML Files (Dimitrov, n.d.)

i. XML file does not have generic predefined markup tags. The user can define new markup tags according to the application. In the proposed model, feature-based markup tags are defined.

ii. XML files have an interpretable format. This makes XML files both human and machine-readable. This property is helpful in the application where data is accessed by both machines and humans.

iii. XML possesses a strong syntax for data storage. This makes implementation more convenient along with preserving the structure of defined markup tags. The feature-based storage of part data ontology is feasible because the XML file syntax can accommodate the feature name and other meta information in its markup tags.

iv. The data-oriented property of XML files helps in retrieving and reusing data at any point in the time

Since a variety of tools are available and accessible to the user, Practically XML is simple for use and used in many applications across industries and different fields. (Ghorpade et al., 2020)(Gil, Y., & Ratnakar, V.; 2002).

2.3 Manufacturing Capacity Parameters

Manufacturing machines are the main elements in the manufacturing ecosystem. Different types of manufacturing machines carry out various operations during the manufacturing of a part. Manufacturing machines are selected for machining the part based on the part’s design, material, and level of complexity. Understanding these factors allows selecting the appropriate capable machines with enough capacity for the machine the required part. This section expands on the study of machine capacity parameters and their importance for the integrative system.

According to the Kalpakjian and Schmid, the manufacturing capacity parameters of a machine are the specifications that describe the capacity of the machine to manufacture a part (Kalpakjian & Steven, 2014). The capability parameters virtually represent the physical manufacturing machine, such a set of information can be called a virtual copy or digital twin of manufacturing machines (Nikolas Theissen, Theodoros Laspas, Karloy Szipka, Andreas Archenti; 2018). Digital twins are widely analyzed and explored for their potential use in cyber-physical manufacturing systems. Many production environments are simulated using the digital twins of physical machines (Theissen et al., 2018). Similarly, dimensioning the information required for creating a digital twin of a physical machine is a research question. An enormous amount of data can be found which is related to the manufacturing machines (Kutin et al., 2019). Currently, a portion of such data is used in applications such as production simulation, predictive maintenance, distance operating, smart manufacturing, etc. The next sections explore a parametric analysis of manufacturing machine capabilities for collecting and storing data to create digital twins.

2.3.1 Capacity Parameters based on Machine Types

Different types of machines carry out different operations. Each of these machines is used for a specific purpose. The turning and milling operations are the most basic subtractive processes that can be performed on raw material to machine a useful component. These processes are often performed on lathe and milling machines, respectively. Another common machine is the machining center which can perform operations of lathe and milling machines. The machining centers can work on multiple axes and along different orientations of the given part depending on the specific needs.

The Lathes are primarily used for machining cylindrical parts. Turning operations are commonly performed on a bar material that rotates on the same axis. Along with the turning operations, a lathe machine can perform various other operations such as facing, cutting, boring, drilling, parting, threading, knurling, straight turning, taper turning, profiling, etc. According to Kalpakjian, the table data such as length and width of the table, chuck diameter capacity, the maximum distance between headstock and tailstock, and tooling information commonly define the capacity of a lathe (Kalpakjian & Steven, 2014). These parameters parameterize the workspace available in the machine to carry out different processes. Moreover, other parameters such as given in Table 1 define the operability of the machine on the given part based on the surface finish requirements and material specifications (Kalpakjian & Steven, 2014) (Ghorpade et al., 2020b).

Table 1. Maximum Workpiece Dimensions for Lathe Machine (Kalpakjian & Steven, 2014)

Machine Tool	Maximum Dimension (m)	Power (kW)	Maximum Speed (rpm)
Lathes (swing/length)			
Bench	0.3/1	<1	3,000
Engine	5-Mar	70	12,000
Turret	0.5/1.5	60	6,000
Automatic Screw Machines	0.1/0.3	20	10,000

Milling machines have diverse capabilities some of them which are similar to the lathes. Although, there is an overlap between lathe and milling machine's operation performance spectrum, a major difference between these two types of machines is, the milling machine operates on non-cylindrical parts. Some of the important operations performed on a milling machine are peripheral milling, slab milling, face milling, end milling, etc. The 5-axis milling machines possess more variability in part manufacturing compared to a conventional 3-axis machine. Using higher axial movement, a 5-axis milling machine is capable of machining complex parts and features with cylindrical orientation. The workspace of a milling machine is defined using the dimensions of the table, tool swing movement, and axis configuration of the milling machine. Similar to the lathe machine, the milling machine contains a spindle that incorporates the power and torque measures that define the operability of the milling machine. Furthermore, like the milling machine, the machining centers possess similar workspace parameters. (Kalpakjian & Steven, 2014). The effect of process parameters such as federates, spindle speed is discussed in the next section.

2.3.2 Literature Review on Capacity Parameter's Effect on Manufacturing Processes

The manufacturing machine's process parameters plays important role in the product formation and accuracy of machining. In milling operations, the important process parameters for the end milling process are cutting speed, depth of cut, and feed rate (Yusup et al., 2012) (Pawar & Rao, 2013) (Muruganandam & Pugazhenthi, 2010). The previous work has shown that these parameters affect the machining time, tool wear during the machining operations, and machining performance measurements (Manna & Bhattacharayya, 2005) (Zainal et al., 2016). Furthermore, during operations performed on a machining center and other milling operations such as slot milling and face milling; selection of cutting speed, depth of cut, and feed rate affects the accuracy and surface finish of the part (Hamdan et al., 2012) (Yusup et al., 2012). Similarly, in lathe operations, cutting speed, depth of cut, and feed rate affect the accuracy and performance of the machine during the machining process (Jasiewicz et al., 2018). The effects of manufacturing process parameters show the importance of these parameters in the machining process. An optimal value of these parameters increases the machining accuracy, surface finish, and reduces the tool wear and machining time.

The dimensional and workspace parameters of manufacturing machines, such as table dimensions, tool movements, and tool holding device dimensions such as chuck and table slots are important for understanding the fitment of the given part into the machine workspace. The compatibility of part and machine largely depends on the fitment of part within the range of machine workspace. For example, the lathe machine has a specified sized chuck and can hold a raw bar material within the limits of the chuck. Which restricts the manufacturing capacity of the machine concerning the part's dimensions. The part and machine compatibility are depended on the part dimensions and machine workspace dimensions along with the range of power and torque ranges of the machine (Li et al., 2016).

For analyzing the availability of capacity parameters on open source websites for the populating machine capacity database, an analysis was conducted on a sample CNC machine manufacturing brand models and their product specifications. The information used to populate the below tables is obtained by examining product specifications from CNC machine brands like Haas, Mazak, and Doosan.

Table 2. HAAS Machine Capacity Parameters (HAAS, 2020)

Availability of Open Source Information of HAAS CNC Machine Models									
Machine Model	**Machine Type**	**Axis Travel**	**Spindle Size**	**Rating Info**	**Spindle Speed**	**Spindle Torque**	**Table Dimensions**	**Feedrate Info**	**Tooling Info**
VF-2	**Mill**	A	A	A	A	A	A	A	A
VR-8	**Mill**	A	A	A	A	A	A	A	A
ST-15	**Lathe**	A	A	A	A	A	NA	A	NA
DS-30Y	**Lathe**	A	A	A	A	A	NA	A	A
VM-3	**Machining Center**	A	A	A	A	A	A	A	A
EC-1600	**Mill**	A	A	A	A	A	A	A	A

Table 3. Mazak Machines Capacity Parameters (Mazak, 2020)

Availability of Open Source Information of Mazak CNC Machine Models									
Machine Model	**Machine Type**	**Axis Travel**	**Spindle Size**	**Rating Info**	**Spindle Speed**	**Spindle Torque**	**Table Dimensions**	**Feedrate Info**	**Tooling Info**
FJV-200	**Mill**	A	A	A	A	NA	A	NA	A
Variaxis C-600	**Mill**	A	A	A	A	A	A	NA	A
QTU-250	**Lathe**	A	A	A	A	NA	NA	NA	A
Quick Turn 100 MS	**Lathe**	A	A	A	A	NA	NA	NA	A
Integrex e-500H-S	**Machining Center**	A	A	A	A	NA	A	NA	A
FF-5000/50	**Mill**	A	A	A	A	NA	A	NA	A

Table 4. Doosan Machines Capacity Parameters (Doosan, 2020)

Availability of Open Source Information of Doosan CNC Machine Models									
Machine Model	**Machine Type**	**Axis Travel**	**Spindle Size**	**Rating Info**	**Spindle Speed**	**Spindle Torque**	**Table Dimensions**	**Feedrate Info**	**Tooling Info**
DNM 4500S	**Mill**	A	A	A	A	A	A	A	A
VC 630/5AX	**Mill**	A	A	A	A	A	A	A	A
Lynx 2100LB	**Lathe**	A	A	A	A	A	NA	NA	A
PUMA GT2600	**Lathe**	A	A	A	A	A	NA	NA	A
VCF 850L II	**Machining Center**	A	A	A	A	A	A	A	A
NHP 5500	**Mill**	A	A	A	A	A	A	A	A

In the above tables ‘A’ represents the model’s given information is available on the website and’ NA’ represents not available (Ghorpade et al., 2020). From the above tables, the product specifications available on open source websites are similar across all three brands. The study was conducted on a sample batch of different models of the mill, lathe, and machining centers.

2.4 Engineering Ontology

The product generates various types of information throughout its life-cycle. This information is generated from different entities of industries ranging from part design data to the product's maintenance log. Such a wide range of knowledge is inter-related with other information generated during the product's life-cycle. For storing this relational product's life-cycle data is valuable for reuse. Moreover, storing and reusing this data often requires a standard methodology (Patil et al., 2005). This formal way is offered by an engineering ontology.

An engineering ontology is a systematic method of storing different engineering entities, their types, and interrelation information for reusing in the next phases of the product's life-cycle (Kathe, 2018). Previously, different research has been done on the engineering ontology. Patil and Dutta proposed Product Semantics Representation Language (PSRL). This proposed ontology provides formal data semantics for product data for PLM (Patil et al., 2005). Another work on ontology proposed by Ruijven defines an ontology for processes involved in systems engineering for establishing a standard environment for projects and information flow (Van Ruijven, 2013).

2.5 Part Dimension and Machine Capability Mapping

The part feature is a generic shape of a physical part element (Sateesh & P, 2017). According to Sormaz and Sarkar, advanced tools such as computer-aided process planning (CAPP) are capable of selecting the processes for the given part based on the part dimensions and the features. The manufacturing processes and sequence of these processes mainly depend on the feature dimensions and the orientation in a given part. Moreover, the milling machine provides an extensive range of different machining operations as discussed in the above section. Hence, a part to be manufactured on a milling machine exhibit a more complex structure and process selection criteria. The automated process selection systems undertake two knowledge bases for deciding the processes to be performed. These criteria are (Sormaz et al., 2018).

- Process selection rules: The process selection rules are based on the individual features of the part. Each feature generates a set of operations. Using these rules generated from such a set of features, a universal set of process sequence rules are finalized for the entire part (Sormaz et al., 2018).
- Machine representation: The machine representations are the set of dimensions and tolerances that can be machined on the given machine (Sormaz et al., 2018).

According to Sormaz and Sarkar, rule-based process selection (RBPS) examines the process capabilities of machines and the feature dimensions and tolerances of parts. The RBPS model maps the machine process capabilities with part features. Figure 9 shows a schematic diagram of part-machine mapping (Sormaz et al., 2018). The above selection criteria are generated based on the given are machine and part data. Each part feature defines the processes and the level of accuracy regarding the machine operation. Similarly, the machine exhibits a set of achievable dimensions and tolerances (Sormaz et al., 2018).

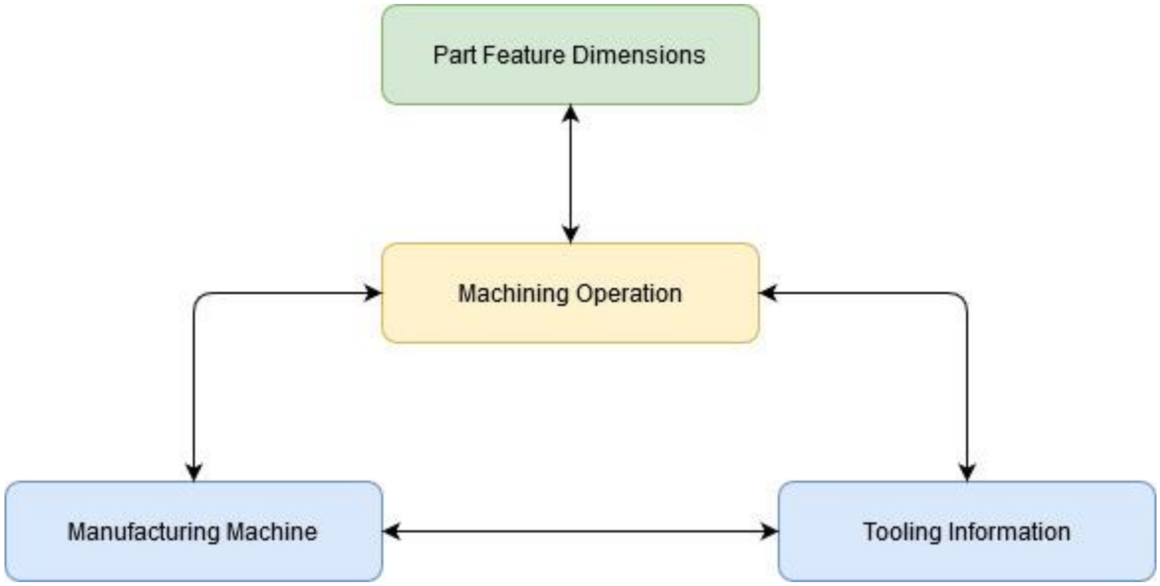

Figure 12. Part-Features and Machine Data Mapping (Sormaz et al., 2018)

The above-specified technique for feature mapping undertakes an individual machine's historic machining performance and the part feature for mapping the processes and part features.

2.6 Active Learning

Training the machine learning model requires a significant amount of data for understanding the hidden patterns and the correlation between predictors and predicting variables. Moreover, for enhancing the model performance, the training set should include the same class distribution as the population dataset. Often acquiring a sample dataset with traits and properties of the actual population is a challenging task. These population properties include predictor class and variable distributions, and predictor class balance. Moreover, even with enough amount of dataset, labeling the observations of the dataset is an expensive and labor-intensive task (Y. Liu, 2004). Hence, to overcome such challenges, an active learning method is being used (Kremer et al., 2014).

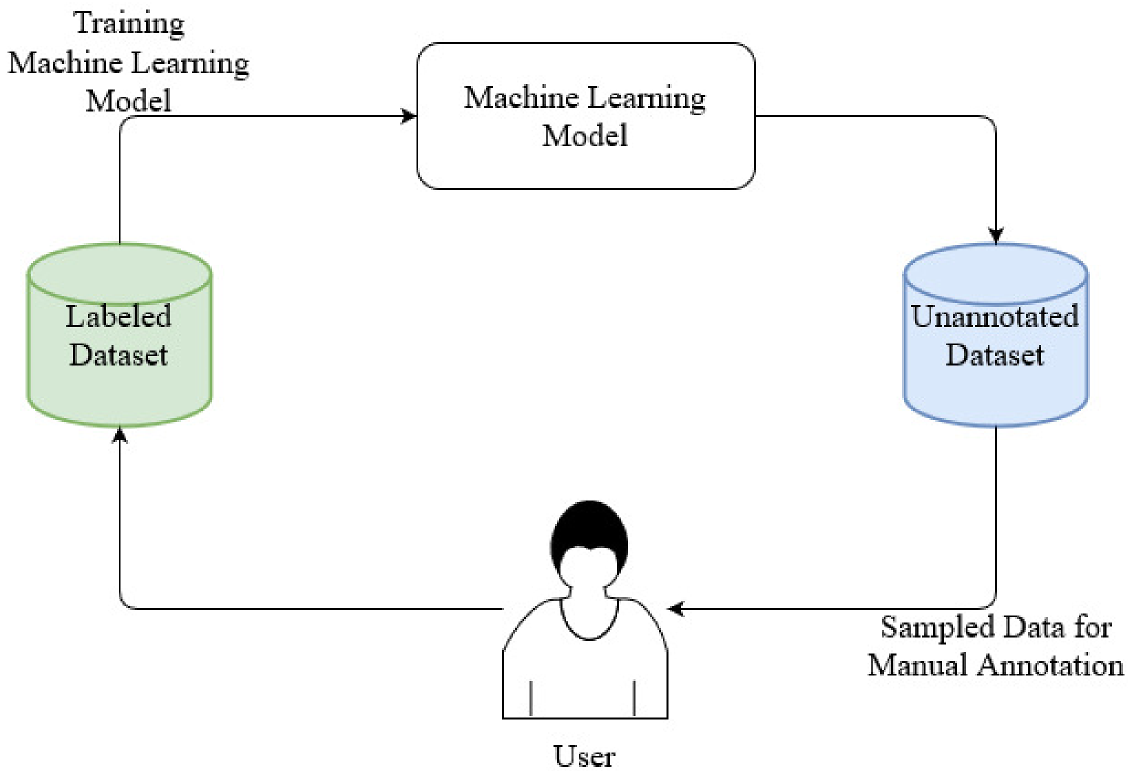

Figure 13. Data Flow in Active Learning Method (Settles, 2010)

Active learning is a machine learning method, which is used for training the machine learning models where a labeled data set is hard and expensive to acquire. A schematic diagram of the active learning method is shown in figure 13. In the active learning approach, a small sample of data is manually labeled (annotated) for the predicting class. Further, this dataset is used as a training set for training a predicting model. Next, based on the performance, and the pattern observed in the training process, another sample is chosen from the data pool and manually labeled. This process is repeated until the machine learning model's performance is improved to the level of acceptance.

Previously, many classification models are trained using an active learning approach. Ying Liu trained the support vector machine (SVM) model on gene expression dataset for cancer classification. The dataset with numerical variables was trained by formulating a binary classification problem. The class distribution in such classification algorithms plays a vital role. The ratio of the positive class (cancer detection) to the negative class (non-cancer detection) of the training set should be the same or significantly close to the population's ratio. In this research, the SVM model trained using active learning outperformed the conventional passive SVM model trained on the same dataset (Y. Liu, 2004).

A research article by Wensong Liu explains the application of active learning for training a random forest classification model for classifying polarimetric features from the Polarimetric SAR images. The image data captured in form of polarimetric SAR images need an intensive human workforce for manually classifying the features from a given image. The application of active learning improved the performance of the classifier along with reducing the manual labeled data requirement (W. Liu et al., 2018).

2.7 Summary of Literature Survey

The summary of the literature survey conducted for understanding the CAD file structure and feature-based storage schema is shown in table 5. In table 5, T1 denotes facial recognition based on CAD file, T2 denotes STEP file structure, and T3 denotes a survey for XML file structure and storage schema.

Similarly, Table 6 shows a summary of the literature survey conducted for understanding the effects of process parameters on machining outcomes. In table 6 T1 Milling machine and process parameters, and T2 Lathe and its process parameters.

Table 5. Summary of Literature Survey for CAD and XML Files

Research Works	T1	T2	T3
Mathematical Representation of Feature Conversion for CAD/CAM System Integration (Gao et al., 2004)	X		
A Novel and Smart Interactive Feature Recognition System for Rotational Parts Using a STEP File (Al-wswasi & Ivanov, 2019)	X	X	
A Methodology for feature extraction and recognition for CAD/CAM Integration using STEP File (Sateesh & P, 2017)	X	X	
A Comparison of (Semantic) Markup Languages (Gil & Ratnakar, 2002)			X
XML Standards for Ontology Exchange (Dimitrov, n.d.)			X

Table 6. Summary of Literature Survey for Machining Process Parameters

Research Works	T1	T2
Evolutionary Techniques in Optimizing Machining Parameters (Yusup et al., 2012)	X	
Parameter Optimization of Machining Processes using Teaching-Learning-Based Optimization Algorithm (Pawar & Rao, 2013)	X	
Influence of Machining Parameters on the Machinability of Particulate Reinforced AI/SiC-MMC (Manna & Bhattacharayya, 2005)	X	
Assistance of Machining Parameters Selection for Slender Tools in CNC Control (Jasiewicz et al., 2018)		X
Glowworm Swarm Optimization (GSO) for Optimization of Machining Parameters (Zainal et al., 2016)	X	

The literature survey conducted for understanding STEP file structure and the entity-based part representation helps in developing an algorithm for dimensional data extraction from STEP files using a feature-based approach. The research articles by Sateesh and Al-wswasi give a significant description of the STEP file structure and the facial representation of the CAD model in the entity-based structure of STEP files.

In the book “Manufacturing Engineering Technology” by Kalpakjian, the significance of machine workspace for machining and part orientation is explained. This explanation is helpful for understanding the importance of a machine’s workspace dimensions for part manufacturing and their effect on the capability for machining. Various research work conducted on manufacturing parameter optimization explains the importance and the effects of manufacturing process parameters on the machining quality and machining time. These references are useful for selecting manufacturing capability parameters. In the research articles by Yusup, Pawar, and Manna, the effects of milling parameters are discussed. Similarly, the research article by Jasiewicz discusses the process parameter selection and optimization for lathe machine. In chapter 4, the manufacturing secondary capacity parameters are selected based on the effect and the importance of these parameters discussed in the above articles.

The research article by Sormaz and Sarkar, explains the rule based-process selection (RBPS) for automating the process selection for part manufacturing. In this article, a machine’s historical knowledge bases are used for understanding the machining pattern and therefore, quantifying the capability. In the proposed model, the machine’s capability is quantified using the capability parameters and the part features using an active learning approach. The pattern explained in this research article helps in understanding the applicability of pattern matching between part features and the machine’s capability parameters.

The previous implementation of active learning approaches can be found in articles by Yang Liu and Wensong Liu. Yang Liu used the active learning approach for classifying cancer genes. The dataset with gene expression data contains gene expression data and possess an internal pattern for detecting cancer. Such an out-of-field classification example is taken as a reference for proposing the active learning approach for classifying the part feature and machine capacity pair as manufacturable or non-manufacturable. In this research article, a combined approach of active learning and support vector machine is used. The support vector machine is a plane-based classifier and has given significant results for classification accuracy. In the research article by Wensong Liu, the active learning approach is combined with the random forest algorithm for polarimetric features from the Polarimetric SAR images. These images are expressed in the feature-based approach and it helps to build the bridge for the implementation of the active learning approach for feature-based part data, where each feature is used for deciding the compatibility with a given machine's capability for manufacturing.

Chapter 3: Research Objective and Scope

3.1 Research Goal

The goal is to establish the process of data collection and storage for developing machine learning models for predictive matching of the manufacturing capabilities of machines with part dimensions. Virtual copies of machine shops are created using machine capacity parameters based on the machine type, brand, and model specifications. Manufacturing machines possess unique applications, each machine is used for a specific type of manufacturing and such information is crucial for selecting the machines for manufacturing the desired part.

3.2 Scope of Peer to Peer Metrological Data Sharing Model

Data is today's oil, and information is power. With recent developments in technology, artificial intelligence (AI) has become an integral part of industries. From oil and gas to healthcare and from entertainment to weather, jobs have been shifting towards smart systems (Chollet, 2018). The manufacturing industry is no different, it is in high demand for high-tech savvy and smart information grids (Hirsch-Kreinsen et al., 2019).

Much data is often wasted every moment across the manufacturing industries. This data contains information on manufacturing processes, lead times, capacity parameters, machine status, etc. The acquisition of such data could enable predictive analytics across the manufacturing sector if suitably analyzed and modeled. Based on a survey conducted for this research of small-medium businesses, most of the manufacturing-related data is not being collected. Hence, we are greatly limited in our ability to use any statistical and machine learning algorithms to make manufacturing more efficient (Waurzyniak, 2015).

Experienced machine operators can judge the machinability or feasibility of making a part on a given machine, based on their skill and experience developed over the years. This task could be comparable to any predictive machine learning task. In such tasks, a well-trained machine learning model identifies the patterns in the predictor variables. The proposed model enables the application of machine learning similarly on capacity parameters and the part design geometrical features. Each machine has parameters that describe its capacity for machining, including power, space, and efficiency.

3.3 Research Boundary

The proposed model for connecting manufacturing capability and part features relies on several data acquisition and storage methods. The manufacturing data needs to be collected and stored to process using advanced data mining techniques.

This thesis research focuses on proposing a model for data collection, data preparation, storage, and methods for linking the manufacturing data and part features. The last section of linking different data utilizes a machine learning approach. Machine learning has shown significant applications in various fields in recent times. But the major drawback of this technique is that it requires a large amount of data for training the models. In this research, we are proposing a framework for collecting and creating a manufacturing database. This research focuses more on the data collection strategies, and less on the implementation of machine learning techniques. The future scope of the research expands on the real-time application of machine learning models and using predictive analytics on the developed database.

Chapter 4: Methodology and Example Implementation

This chapter discusses the proposed model for peer to peer metrological data integration system. The system's operations are grouped into the steps. Each step is explained using examples. The integration of part data with manufacturing capability data is carried out using machine learning. This thesis does not focus on the training of the machine learning model, whereas more focus is given to the collection, storage, and processing of the data for the system integration.

4.1 Proposed Model

The preceding sections have discussed the fundamental need and background concepts for data collection and integration system from different manufacturing elements. Figure 14 shows the schematic model of the proposed data collection and integration of manufacturing capability with part CAD data. The specified data inherently possess different data types and the integration of these two attributes of manufacturing systems to create smart grids in manufacturing systems is a challenging task. The proposed model's data flow is shown in figure 14.

The proposed model has bidirectional data inflow. The manufacturing capability data is collected and stored in the form of a manufacturing database. Such a database gives the virtual copy of machine-shops representing the machine shop's manufacturing capability. On the other hand, part designs are abundantly available in various CAD formats. These CAD files vary in their format based on the CAD system used for design. For coherence of the CAD data, a universal CAD neutral file format is used as input for the system. 3D CAD models are represented in STEP AP203 native format. Using the proposed algorithm, the part features are further extracted from STEP files and stored in the proposed format in XML documents. At last, a systematic data integration schema is proposed for training machine learning models to map the patterns between parts and machines.

Based on the data inflow and processing, the research plan is divided into three independent sections.

I. Manufacturing capability or virtual machine-shop section,

II. CAD section, and

III. Feature matching section.

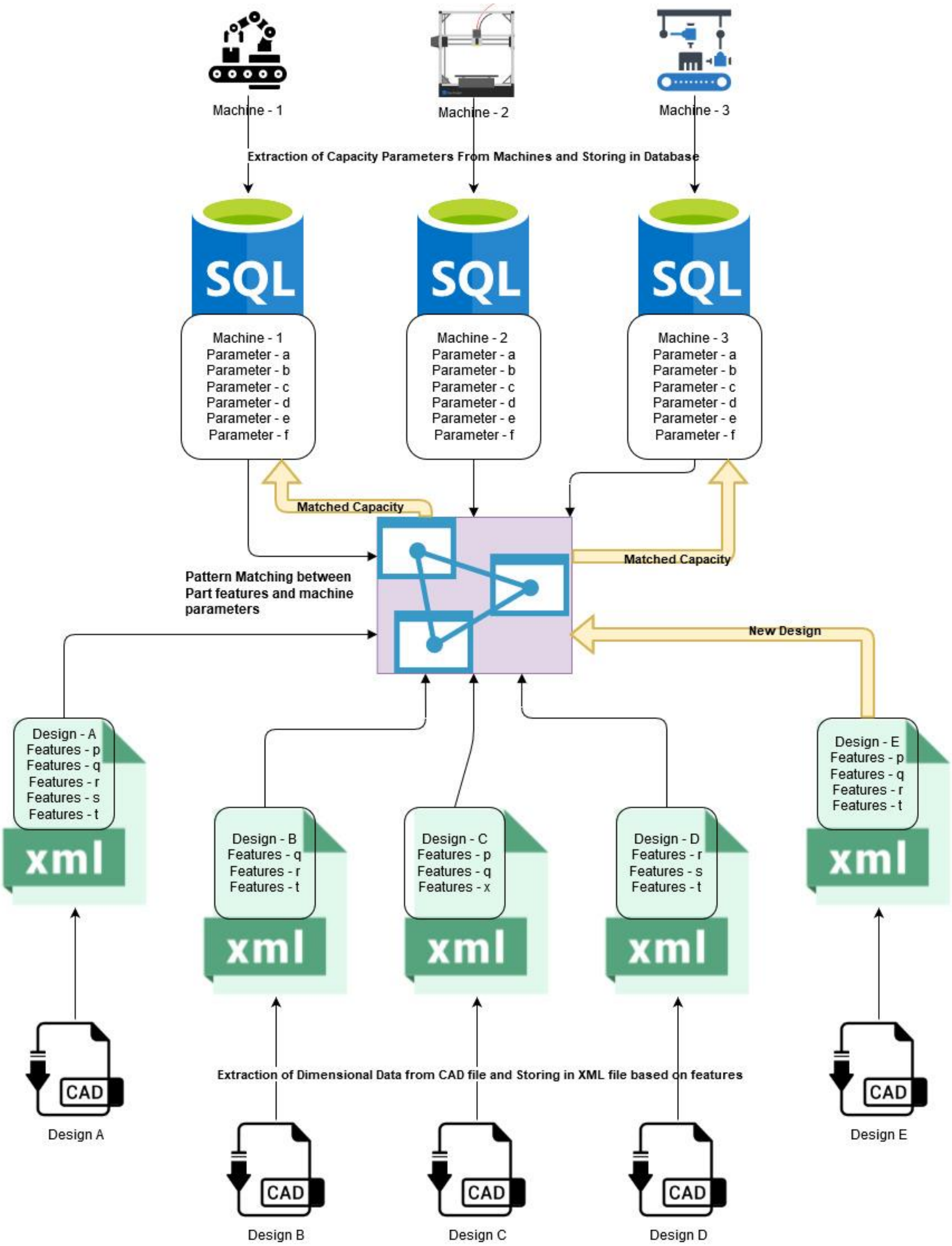

Figure 14. Schematic Structure of Proposed Model for Peer to Peer Metrological Data Integration

Section i and ii are focused on the data acquisition and storage of manufacturing capability data and CAD data, respectively. On the other hand, the matching section deals with linking these two different data streams for recognizing the pattern of relationships between manufacturing capacity parameters and CAD features.

The research study is more focused on the data acquisition and storage of the data. However, a conceptual pattern matching model is proposed in section iii.

I) Manufacturing Capabilities

The machines are the main elements in the manufacturing ecosystem. Hundreds of different machine types and thousands of different variants can be found across the stretch of the manufacturing industry. But each machine has its capability specifications to perform tasks. These capability parameters are important to decide the manufacturability of a given part on the given machine. The capacity of a machine is represented using several different parameters, and every parameter is important for understanding the compatibility of a part to be machined and the physical manufacturing machine. This set of machine capability parameters forms the manufacturing machine's virtual copy. In figure 14. these parameters are generically denoted as "a" through "f". The parameters are the machine specifications defined by the machine manufacturer. As seen in chapter 2, these parameters play a significant role in the part fitment in the machine's workspace, machining process's accuracy, and machining time. In this thesis, the machine's capacity parameters are assumed to be mutually independent, operational wear, deformation of machine parts, and deviation of machine's performance over the service period are assumed to be null. However, the inclusion of these effects is suggested in the last chapter for future study.

Chapter 2 reported the extensive study on Capacity parameters for lathes, milling machines, and machining centers. These capacity parameters are categorized into two classes.

i) Primary parameters:

The primary capacity parameters are the dimensional parameters of machines. The workspace of the machine is defined by its primary parameters. As shown in Table 7, each machine's workspace is quantified using the possible maximum space of part holding and moving tooling capacities, and tool swinging dimensions.

For example, the lathe machine's primary workspace parameters contain axis travel space in X and Z direction, and Spindle parameters such as spindle nose type and bore diameter. Primary tool parameters of lathe machines are turret specification, max tool size, a tool to chip dimension, etc. Table 7 lists the primary parameters of three types of machines: Milling machine, Machining center, and Lathe.

ii) Secondary parameters:

The secondary parameters are the operational parameters of manufacturing machines. These parameters define the operational capacities of the machine such as mechanical power, feed rates, and lubrication. These parameters are important for defining the surface finish, machining time, and machine accuracy. Moreover, tool life largely depends on the selection of these parameters during machining.

The summarization and classification of the capacity parameters of formerly analyzed machines help analyze the direct synchronization between capacity parameters and dimensions. In summary table 7, the field with text applicable represents the availability of the parameter in the specific machines, whereas NA indicates the unavailability. Some of the parameters are only available in special cases, such as with the limited axis operations; such parameters and their machine type are represented by the yellow field.

Table 7. Summary of Capacity Parameters

Category	Sub-Category	Machine Capacity Parameters	Machines		
			Non- Rotational Machines		Rotational Machines
			Milling Machine	Machining Center	Lathe Machine
Primary	Axis Travel	X, Y, Z Dimensional Data	Applicable	Applicable	Only X, Z
	Table Data	Length	Applicable	Applicable	Applicable
		Width	Applicable	Applicable	NA
		Number of T-Slots	Applicable	Applicable	NA
		Max Weight	Applicable	Applicable	NA
	Spindle and Chuck	Chuck Size	NA	NA	Applicable
		Max Cutting Diameter	NA	NA	Applicable
		Max Cutting Length	NA	NA	Applicable
		Spindle Nose to Table	Applicable	Applicable	NA
		Spindle Bore Diameter	NA	NA	Applicable
		Spindle Nose Type	NA	NA	Applicable
	Turret	Number of Tools	Applicable	Applicable	Applicable
	Tool Data	Type	Applicable	Applicable	Applicable
		Capacity	Applicable	Applicable	Applicable
		Max Tool Diameter	Applicable	Applicable	Applicable
		Max Tool Weight	Applicable	Applicable	NA
		Tool-to-Tool	Applicable	Applicable	NA
		Chip-to-Chip	Applicable	Applicable	NA
Secondary	Feedrates	Max Cutting	Applicable	Applicable	NA
	Spindle	Max Rating	Applicable	Applicable	Applicable
		Max Speed	Applicable	Applicable	Applicable
		Max Torque	Applicable	Applicable	Applicable
	Other	Lubrication	Applicable	Applicable	Applicable
		Cooling	Applicable	Applicable	Applicable

II) Computer-Aided Design (CAD) Files

The part design files are stored in CAD formats. There are various formats available based on the software used to create the drawings. The proposed model uses a universal native CAD STEP AP203 format file for dimensional data extraction. The extracted dimensions are stored in an eXtensible Markup Language (XML) file format, using geometrical feature-based logic. The part's feature is the generic shape of the part. In figure 14. generically these features for each design are denoted. For example, design A has features (p, q, r, s, t), whereas design B has features (q, r, t). The part designs are made up of a combination of geometrical features or geometrical faces. As seen in chapter 2, the geometric features are important segments of any given design file.

Chapter 2 reported an extensive study on STEP file structures by exploring a tree-based hierarchical structure. The extraction of actual dimensions of any part can be carried using the explained structure of STEP files. The STEP file is compiled with two types of the part's geometric information: face orientation information and face location coordinate points in the CAD workspace environment. For extracting the dimensions these two types of data need to be systematically processed. These extracted dimensions are specifically related to the faces (features) of the parts. These dimensions can be extracted using a specially developed algorithm. The proposed algorithm as shown in figure 15, takes the STEP file as input and outputs the face dimensions including radiuses of circular sections. The flowchart explaining the proposed dimension extraction algorithm is shown in figure 15. Each advance face is to be processed to get the Cartesian points of the borders of the face. This involves the stepwise processing of each STEP file entity of the advanced face. The main STEP file entities as discussed in chapter 2, are FACE_OUTER_BOND, EDGE_LOOP, ORIENTED_EDGE, and EDGE_CURVE. Each face has multiple edges, and these numbers of edges are equal to the number of Edge Curves in the STEP file for the given face. Each Edge Curve has three attributes, in which two are vertex points of a given edge, and the third attribute discloses the nature of the edge, which is either circular or linear.

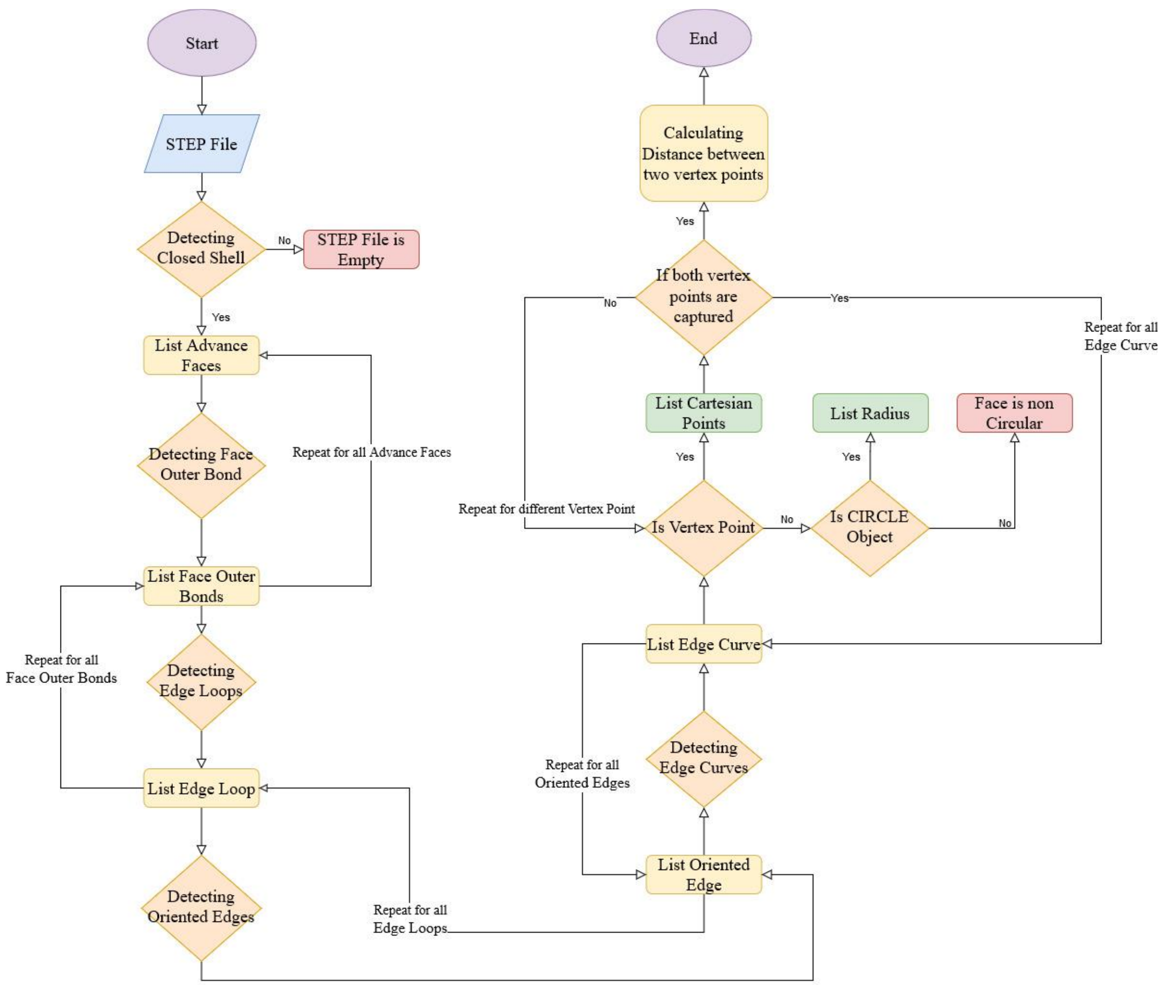

Figure 15. Flowchart of Data Extraction Algorithm

The circular attribute discloses the radial dimension of the edge (radius). Since the linear edge often does not contain any important information within it, we can extract the linear dimensions of the edge using Cartesian points. Each vertex point contains the

Cartesian point of the edge. Hence, all vertex points of the edge are used to calculate the distance between them.

For calculating the distance, the three-dimensional Euclidean distance formula as given below is used (ScienceDirect, 2020). If P_1 and P_2 be Cartesian points of two adjacent vertex points (V_1, V_2) of edge curve (E_c), then d(P_1, P_2) is the linear distance between vertex points. This linear distance between co-ordinate points (x_1, y_1, z_1) and (x_2, y_2, z_2) is given as:

$$d(P_1, P_2) \quad = \quad \sqrt{((x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2)}$$

The extracted dimensions are stored in an XML file using feature-based logic.

III) Feature matching section

The third section integrates the data stored in the XML file and manufacturing capability database. For feature matching active learning approach is proposed. The Active learning approach is fundamentally based on machine learning techniques. A supervised machine learning model is trained using known data for predicting the probability of manufacturability of a certain part at a given machine shop or machine.

As discussed in chapter 2, active learning is a supervised or semi-supervised learning method. Active learning is trained on a small sample of the unlabeled dataset by labeling the sample with the respective class in case of a classification problem. This approach is suitable when the annotation for each observation in the dataset is expensive or not humanly possible (Settles, 2010). In the proposed framework, the data generated needs to be labeled as machinable or non-machinable and this is expensive for the industry. Hence, using an active learning approach is suitable for the proposed framework.

The trained model produces the probability of a given pair of part features and machine capacity observation to be machinable and non-machinable. Further, a cutoff probability point is used for distinguishing the classes. In this trainable data, each machine data is linked with several part data, and such pairs are further individually labeled as machinable or non-machinable. This pre-labeled data is used for training and tuning the machine learning model. Since each given pair of the machine and part features can be either machinable or non-machinable, therefore, each such pair is given one label. Therefore, in this approach, the pattern matching is considered as a binary classification single label problem.

Since this approach requires training a machine learning model on the labeled pairs of part dimension data and manufacturing capacity data, the implementation of the approach is outside of the scope of current research. Chapter 6 discusses future research using this approach.

4.2 Steps Involved in Data Extraction

The data discussed in the above sections are not readily available, and if it is then often is not in a proper format for use. Hence, a rigid and general data extraction process needs to be defined.

The following steps extensively collect, preprocess, and stores the data into a reusable format.

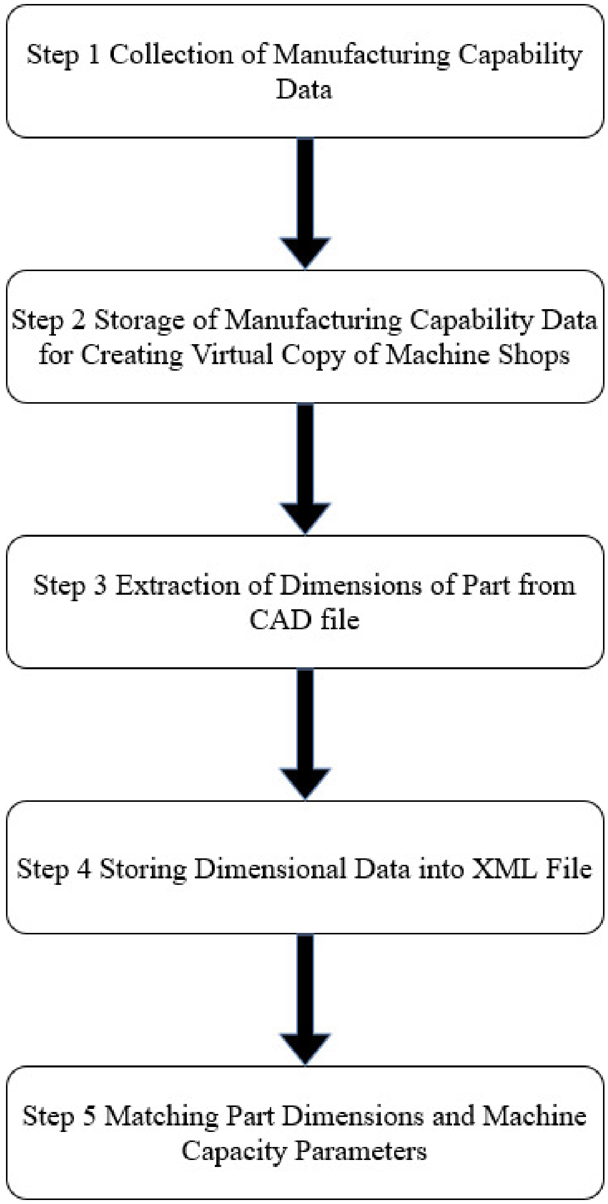

Figure 16. Steps in Proposed Framework

4.2.1 Step 1. Collection of Manufacturing Capability data

Across the manufacturing industries capability data is not stored in form of machine specification. The nearest easily available information about machines is their brand and model information. Moreover, as discussed in chapter 1, small-scale manufacturers are reluctant to share their data. But the generic information such as machine quantity, types, brands, and model numbers can be collected. Other specific information of machines can be collected from open web-sources. In this thesis, using the brand and model name, capability data is collected from open websites. Many brands offer open information about machine capacities. The machine capacities vary based on the model; hence, the model number is used for collecting the specifications.

Table 8. Arbitrary Machine Shop's Machine Configurations

	Mill Machine	**Lathe Machine**
Machine Shop A	Haas VF3	Doosan Puma 4100LB
	Haas VF 2-TR	Haas ST-35
	Haas Mini Mill	

For demonstration, as shown in Table 8; an arbitrary machine-shop with a set of machines is considered as an example. The data shown in table 8 is generic and can be made available from machine shops since it does not involve sensitive information of any specific machine shop or machines. The capability data is collected using web scraping from the given machine's brand specifications. The machine specifications are available on the open-source websites of the respective manufacturers. The result of the survey done on the availability of specification data on the websites of three different CNC manufacturers; namely, HAAS, Mazak, and Doosan are shown in chapter 2. The data for the above-sampled machine shop is acquired from their respective manufacture's specification sheets. As shown in table 8, the example machine shop has two types of manufacturing machines. In each type, different brands and models are installed. Using the machine specifications, a virtual copy of this machine shop in the terms of machine capability is generated.

Table 9. Representation of Arbitrary Machine Shop using Machine Capacity Parameter

Machine-Shop A							
Category	Sub-Category	Machine Capacity Parameters	Machines				
			Non- Rotational Machine Type			Rotational Machine Type	
			Milling Machine			Lathe Machine	
			Haas VF 3	Haas VF 2-TR	Haas Mini Mill	Doosan Puma 4100L	Haas ST 35
Primary	Axis Travel	X, Y, Z Dimensional Data (in)	40x20x25	30x16x20	16x12x10	9.4x52.5	9.4x52.5
	Table Data	Length (in)	48	36	28.75		
		Width (in)	18	14	12		
		Number of T-Slots	5	3	3		
		Max Weight (lb)	3500	3000	500		
	Spindle and Chuck	Chuck Size (in) *				18	12
		Max Cutting Diameter (in) *				21.7	15
		Max Cutting Length (in) *				82.36	32.5
		Spindle Nose to Table (in)	in:4.2 / max: 29	min:4.0 / max:24.0	min:4.0 / max:14.0		
		Spindle Bore Diameter * (in)				5.2	4.62
		Spindle Nose Type *				BMT-KEY	A2-8
	Turret	Number of Tools				12	12
	Tool Data	Type	Carousel	SMTC	Carousel		
		Capacity	20	30+1	10		
		Max Tool Diameter (in)	3.5	2.5	3.5		
		Max Tool Weight (lb)	12	12	12		
		Tool-to-Tool (s)	4.2	2.8	4.2		
		Chip-to-Chip (s)	4.5	3.6	5		
Secondary	Feedrates	Max Cutting (ipm)	650	650	500		
	Spindle	Max Rating (hp)	30	30	7.5	50	40
		Max Speed (rpm)	8100	8100	6000	2000	3200
		Max Torque (ft-lbf)	90	90	33	2419	425
	Other	Lubrication	Air/Oil Injection	Air/Oil Injection	Greased Packed		
		Cooling (gal)	Liquid: 55	Liquid: 55	Air Cooled: 40	Liquid	Liquid

Table 9 shows the capability dataset extracted using web scraping using the model numbers of the machine. The virtual copy of machine shops is a virtual representation of the machine shop's capability in terms of the machine's capacity parameters.

4.2.2 Step 2. Storage of manufacturing capability data for creating a virtual copy of machine shops

The extracted data from manufacturing machines are further stored in respective data storage systems. A dedicated entity-based database storage system is designed for storing the manufacturing capability data. The attributes of the database vary based on the type of machines. A schematic representation of the database for different machine types is given in figure 17.

In the proposed model, an entity-relationship database is used for storing the parameters (Silberschatz et al., 2011). In this storage system, each machine with a unique machine ID is stored along with the machine shop ID. The unique ID helps in tracking appropriate machines and their respective machine shops.

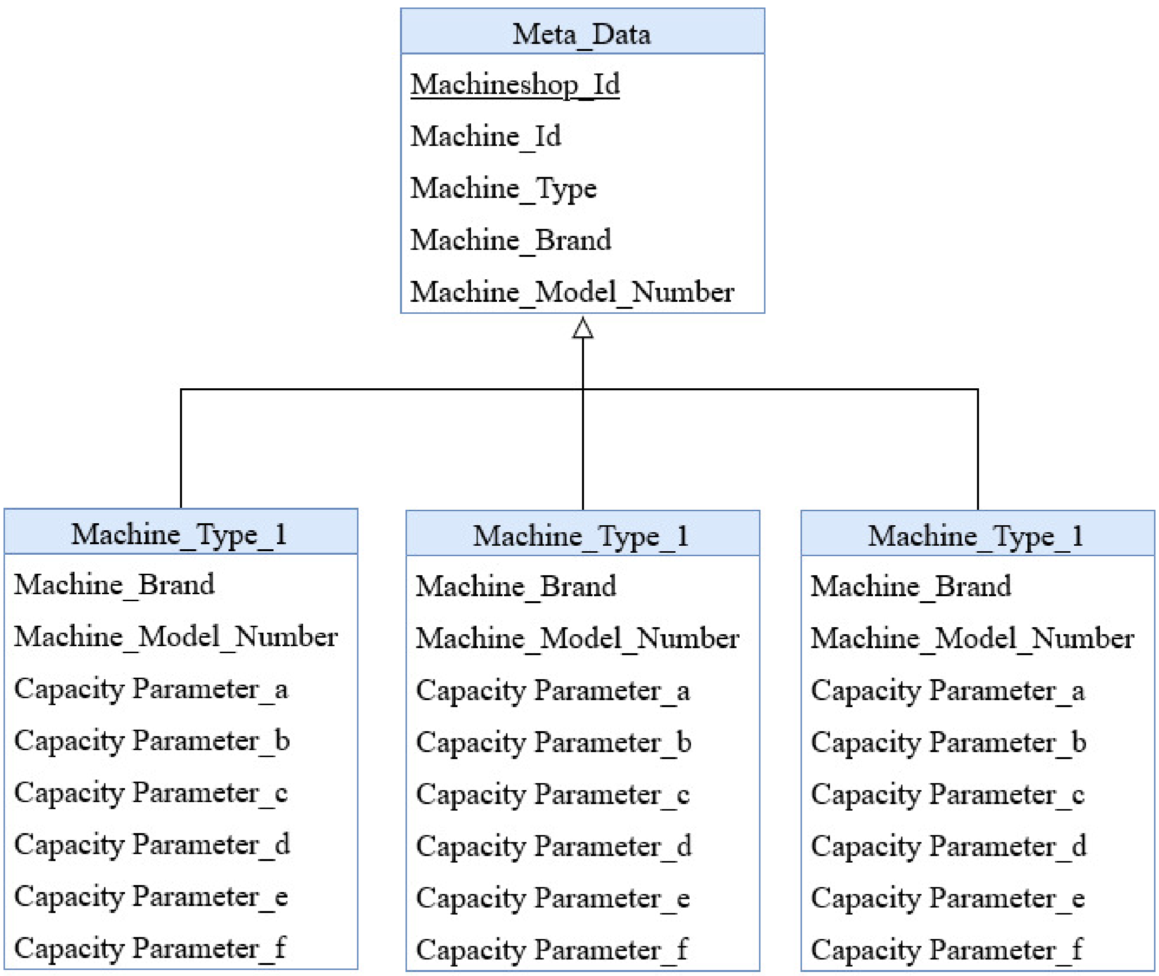

Figure 17. Entity-Relationship Storage Schema for Capacity Parameters

Moreover, each machine further contains meta-information such as machine type, machine brand, and machine model number. For an illustration of the manufacturing storage, the arbitrary machine shop data extracted (collected) in step 1 is stored in respective tables as shown in Figures 18 and 19. Each machine type has specified data fields according to the type of machine and respective capability data is stored in the database. The tables given in figure 17 shows the individual machine type database tables. As discussed earlier, the parameter considered is generic for each given machine type and all machines are considered non-customized.

100 %

Results Messages

	Model_Number	Axis_Travel_X	Axis_Travel_Y	Axis_Travel_Z	Table_Length	Table_Width	Number_of_T_Slots	Table_Endurable_Weight	Minimum_Nose_Table_Length	maximum_Nose_Table_Length	Tool_Type	Tool_Capacity
1	Mini Mill	16.00	12.00	10.00	28.75	12.00	3	500.00	4.00	14.00	Carousel	10
2	VF 2-TR	30.00	16.00	20.00	36.00	14.00	3	3000.00	4.00	24.00	SMTC	31
3	VF 3	40.00	20.00	25.00	48.00	18.00	5	3500.00	4.20	29.20	Carousel	20

Messages

Maximum_Tool_Diameter	Maximum_Tool_Weight	Tool_Tool_Length	Chip_Chip_Length	Tool_Taper	Max_Cutting_Feedrate	Max_Spindle_Rating	Max_Spindle_Speed	Max_Spindle_Torque	Luberication_Type	Cooling_Type
3.50	12.00	4.20	5.00	40.00	500	7	6000	33	Greased Packed	Air Cooled
2.50	12.00	2.80	3.60	40.00	650	30	8100	90	Air/Oil Injected	Liquid
3.50	12.00	4.20	4.50	40.00	650	30	8100	90	Air/Oil Injected	Liquid

Figure 18. CNC Milling Machine Capacity Parameter SQL Table

100 %

Results Messages

	Model_Number	Axis_Travel_Y	Axis_Travel_Z	Chuck_Size	Max_Cutting_Diameter	Max_Cutting_Length	Spindle_Bore_Diameter	Spindle_Nose_Type	Tool_Capacity	Max_Spindle_Rating	Max_Spindle_Speed	Max_Spindle_Torque
1	Puma 4100L	9.40	2.50	18.00	21.70	82.36	5.20	BMT-KEY	12	50	2000	2419
2	ST 35	9.40	52.50	12.00	15.00	32.50	4.62	A2-8	12	40	3200	425

Figure 19.CNC Lathe Machine Capacity Parameters SQL Table

The capability data of the sample machine shop described in step 1 is stored in Microsoft SQL (MSSQL) database as shown in Figures 18 and 19. Figure 18 shows the tabular storage of milling machine capacities and figure 19 shows the tabular storage of lathe in the SQL database system. In these tables, Model_Number acts as the primary key.

4.2.3 Step 3. Extraction of dimensions of Part from CAD file

As discussed in previous sections, CAD data is available in various formats and the dimensional data is extracted in the model using the proposed algorithm. For the demonstration, an example part of the connector is shown in figure 20. The connector part is assumed to be made of metallic material. The study of specific material types of part is outside of the scope of this thesis.

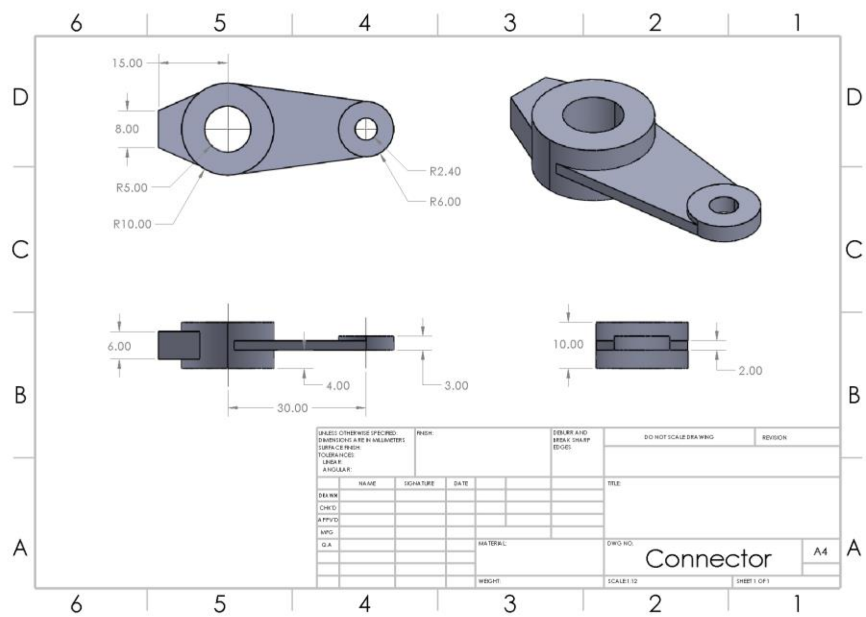

Figure 20. Connector Part Design

A Python programming language-based application is developed for dimension extraction. Using the algorithm all dimensions of the part are extracted. A screenshot of the output is shown in figure 21. The face shown in output is a semi-cylindrical internal face of through-hole with a radius of 2.40 inches and a length of 3.00 inches. The face is highlighted in figure 22. Face outer bond 315 in figure 21, represents the unique integer of the given face in the STEP file. The proposed algorithm uses the same unique integer for representing the face in the extracted output. This numbering of the face helps in tracking the dimensions down and linking the faces with adjacent faces for defining complete features.

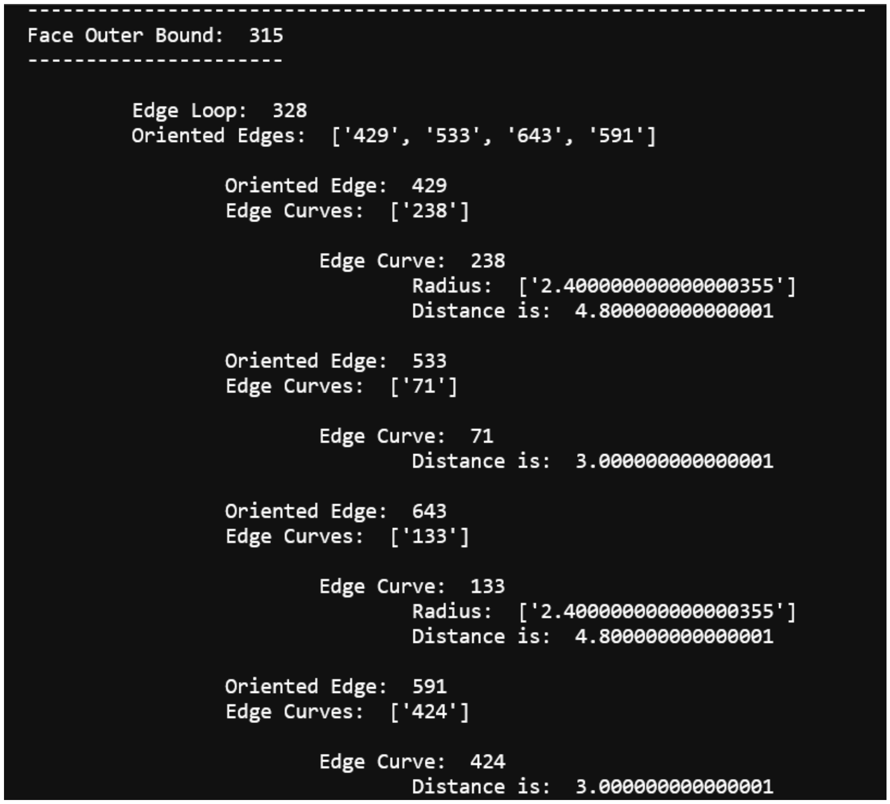

Figure 21. Output of Dimensional Data Extraction Algorithm for Connector Part

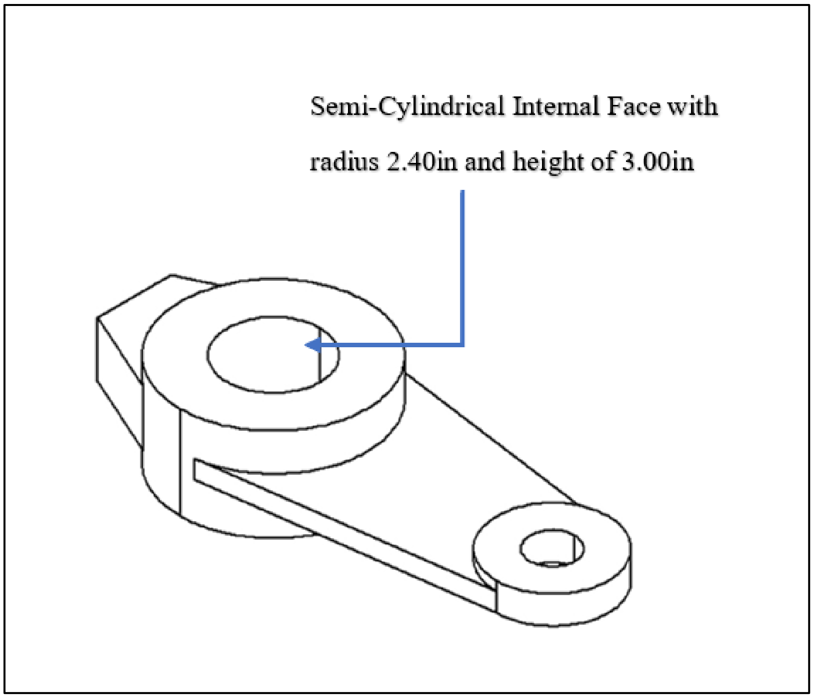

Figure 22. Internal Face of Connector Part

The extracted dimensions are further stored into an XML file in feature-based logic. Feature-based logical data structure for storing dimensional data fulfills the application requirements. Since most face edges are shared with adjacent faces, face level hierarchy helps in reducing the redundancy of dimensional data.

4.2.4 Step 4. Dimensional data storage into the XML file

Extracted dimensional data from the CAD file as shown in Step 3, is stored into a feature-based logical structure in the XML file. The part storage structure for the XML file is explained in the following section.

Discussion on XML Structure for Parts Ontology

In the XML file syntax, elements are the basic components. These elements contain data bounded by the markup tags. The data stored in these elements are represented by the markup tags and stored data can be numeric or text (Abiteboul et al., 2014). In the XML terminology, the information within each element is called content. Content can have again a sub-element or data stored in it. If the content of an element or sub-element contains again a sub-element, then such data storage can be called nested storage. This nested storage helps store data in a hierarchical structure. Figure 23 shows a proposed structure describing a part.

```
<Assembly_Name: __________>
        <Component_1:>
                <Feature_1:>
                        <Dimensions_Circular Radius = _____ />
                        <Dimensions_Linear Length = _____ />
                </Feature_1:>


                <Feature_2:>
                        <Dimensions_Circular Radius = _____ />
                        <Dimensions_Linear Length = _____ />
                </Feature_2:>
        </Component_1:>

        <Component_2:>
                <Feature_1:>
                        <Dimensions_Circular Radius = _____ />
                        <Dimensions_Linear Length = _____ />
                </Feature_1:>


                <Feature_2:>
                        <Dimensions_Circular Radius = _____ />
                        <Dimensions_Linear Length = _____ />
                </Feature_2:>
        </Component_2:>

</Assembly_Name: __________>
```

Figure 23. Snapshot of XML Storage Schema

The expression <Assembly_Name:> and </ Assembly_Name:> are the markup tags for main element. Each XML file represents one assembly. Assembly element further has sub-elements named as <Component_1>, <Component_2>, <Component_3> and so on. These components are basic components of a given assembly. If the extracted dimensions represent a single component and not an assembly, then the main assembly element will contain only one component named as <Component_1>. Each component sub-element stores another sub-element named 'Features'. Here, <Feature_1 :> and </Feature_1:> are start and end markup tags, respectively. These are also called markups. Each feature contains a unique and separate tag. The tag number is given based on the unique integer (line number) where the specific feature is represented in the STEP file. Further, feature sub-element stores, the dimensions of the given feature as sub-elements of features. The start and end tag for such sub-elements can be of two types ***<Dimensions_Linear Length* = _____ />** and ***<Dimensions_Circular Radius* = _____ />**. The former sub-element represents linear dimensions. Whereas later represents radial dimensions.

Each dimensional sub-element stores the data, this data can be numerical or text. In the proposed structure, numeric dimensional values are stored in the sub-elements. For example: <Dimensions_Linear Length = 20 />.

The data format for XML file

Data storage in the XML file is an important factor in this model. An XML file is a document in which entire assembly data is being stored. Moreover, all future updates can take place through this XML file. In the proposed database format, each assembly is considered as an object which is identified by the 'assembly key'. This key encompasses entire assembly information. Under such assembly, various components can exist. Each component has a unique component key. This component key further has various features.

Each feature is recognized by a feature key and the feature key contains specific dimensioning information of the component. The features are automatically recognized by the above algorithm explained in step 3 and based on the feature the XML file creates a new element for respective features.


```xml
<?xml version="1.0"?>
- <Component_1>
  - <Feature_315>
      <Dimensions_Circular Radius="2.400000000000000355"/>
      <Dimensions_Linear Length="3.000000000000001"/>
      <Dimensions_Circular Radius="2.400000000000000355"/>
      <Dimensions_Linear Length="3.000000000000001"/>
    </Feature_315>
  - <Feature_209>
      <Dimensions_Circular Radius="6.000000000000000888"/>
      <Dimensions_Linear Length="1.0000000000000018"/>
      <Dimensions_Linear Length="1.9999999999999991"/>
      <Dimensions_Circular Radius="6.000000000000000888"/>
      <Dimensions_Linear Length="1.9999999999999991"/>
      <Dimensions_Linear Length="1.0000000000000018"/>
    </Feature_209>
  - <Feature_93>
      <Dimensions_Circular Radius="5.000000000000000000"/>
      <Dimensions_Linear Length="10.0"/>
      <Dimensions_Circular Radius="5.000000000000000000"/>
      <Dimensions_Linear Length="10.0"/>
    </Feature_93>
  - <Feature_191>
      <Dimensions_Linear Length="1.9999999999999991"/>
      <Dimensions_Circular Radius="10.00000000000000000"/>
      <Dimensions_Linear Length="1.9999999999999991"/>
      <Dimensions_Circular Radius="10.00000000000000000"/>
    </Feature_191>
  - <Feature_239>
      <Dimensions_Linear Length="10.0"/>
      <Dimensions_Circular Radius="10.00000000000000000"/>
      <Dimensions_Linear Length="10.0"/>
      <Dimensions_Circular Radius="10.00000000000000000"/>
    </Feature_239>
```

```
- <Feature_43>
      <Dimensions_Circular Radius="5.0000000000000000000"/>
      <Dimensions_Circular Radius="5.0000000000000000000"/>
  </Feature_43>
- <Feature_623>
      <Dimensions_Circular Radius="5.0000000000000000000"/>
      <Dimensions_Circular Radius="5.0000000000000000000"/>
  </Feature_623>
- <Feature_548>
      <Dimensions_Linear Length="10.0"/>
      <Dimensions_Circular Radius="5.0000000000000000000"/>
      <Dimensions_Linear Length="10.0"/>
      <Dimensions_Circular Radius="5.0000000000000000000"/>
  </Feature_548>
- <Feature_563>
      <Dimensions_Linear Length="28.913664589601918"/>
      <Dimensions_Linear Length="1.9999999999999991"/>
      <Dimensions_Linear Length="28.91366458960192"/>
      <Dimensions_Linear Length="1.9999999999999991"/>
  </Feature_563>
- <Feature_277>
      <Dimensions_Linear Length="28.91366458960192"/>
      <Dimensions_Linear Length="1.9999999999999991"/>
      <Dimensions_Linear Length="28.91366458960192"/>
      <Dimensions_Linear Length="1.9999999999999991"/>
  </Feature_277>
- <Feature_611>
      <Dimensions_Circular Radius="6.0000000000000000888"/>
      <Dimensions_Linear Length="28.91366458960192"/>
      <Dimensions_Circular Radius="10.000000000000000000"/>
      <Dimensions_Linear Length="28.91366458960192"/>
  </Feature_611>
- <Feature_64>
      <Dimensions_Circular Radius="2.4000000000000000355"/>
      <Dimensions_Circular Radius="2.4000000000000000355"/>
  </Feature_64>
```


Figure 24. XML File Output of Connector Part

For demonstrating the data structure and the storage schema, data extracted from Step 4 is used for creating an XML file. Figure 24 shows the snap of an XML file. The complete file is shown in the appendix.

As seen in figure 20, the connector part is a single component and not an assembly. Hence, the XML file has the only component_1. Within component_1, various features are given. Each feature is represented by the faces that the feature is made of. For instance, the internal face has shown in figure 22 is represented in Feature_315. The hole (internal face) has a radius of 2.4 inches and a depth of 3 inches. Every feature is represented using a similar representation form. This XML file is generated using an extension for the algorithm. In this connector example, python language is used for generating the XML file after extracting the feature-based dimensions from the STEP file.

4.2.5 Step 5. Matching part dimensions and machine capacity parameters

The fifth step is the integrative step for acquired manufacturing capacity parameter data and extracted part dimensional data. As discussed, an active learning approach is proposed to be used for matching the pattern between two distinct datasets. This step gives a method for creating structured data set using extracted dimensions and manufacturing capabilities.

The manufacturing capability dataset is stored in numerical and categorical formats. Both of these are supported by machine learning algorithms and certain standard preprocessing may or may not be applicable depending on the machine learning algorithm chosen. Different machine learning algorithms can be used for training the data. As the scope of this thesis, two algorithms are discussed as given below.

i) Random Forest: The random forest is a tree-based algorithm. A set of trees are trained for the given observation and the outcome of each tree is compared to give the final probability of the classes. Moreover, the random forest is a suitable classification algorithm for a dataset with outliers and categorical variables (Kuhn & Johnson, 2013).

ii) Support Vector Machine: The support vector machine is another classifier commonly used for classification problems. The SVM algorithm helps in understanding the hidden pattern in the dataset. SVM performs well on classification algorithms. The hyperplane separates the classes with the objective function of maximizing the distance between classes (Kuhn & Johnson, 2013).

On the other hand, dimensions are put into the XML structure. As discussed in earlier steps, the XML file can have multiple features depending on the complexity of the part. This creates non-regularity in the dimensional space of part dimensional data. For instance, the connector example has 20 faces, whereas a simple cylinder may have only 4 faces. This illustrates that every part has a different number of faces depending on the complexity of the part. This potentially limits the machine learning approach since this will create unbalanced dimensionality for a training set of different parts. Moreover, some of the part profiles (features) may not be as important. This is because the proposed data structure and storage system indulges in the direction of matching the dimensional constraints between part dimensions and machine workspace. Therefore, only the dimensions which are larger and create a discrepancy with the machine workspace plays an important role.

To navigate the challenge of non-regular dimensionality, and considering the scope of the proposed dataset, a formatted, and compatible variable selection is done by only considering the largest features from the XML file. The fitment of the part into the machine workspace depends on the larger features compared to the smaller features of the part. Such as in the given connector part, the larger face has 30inches in length as shown in figure 20, and this feature is important for deciding the fitment of the given part into the machine.

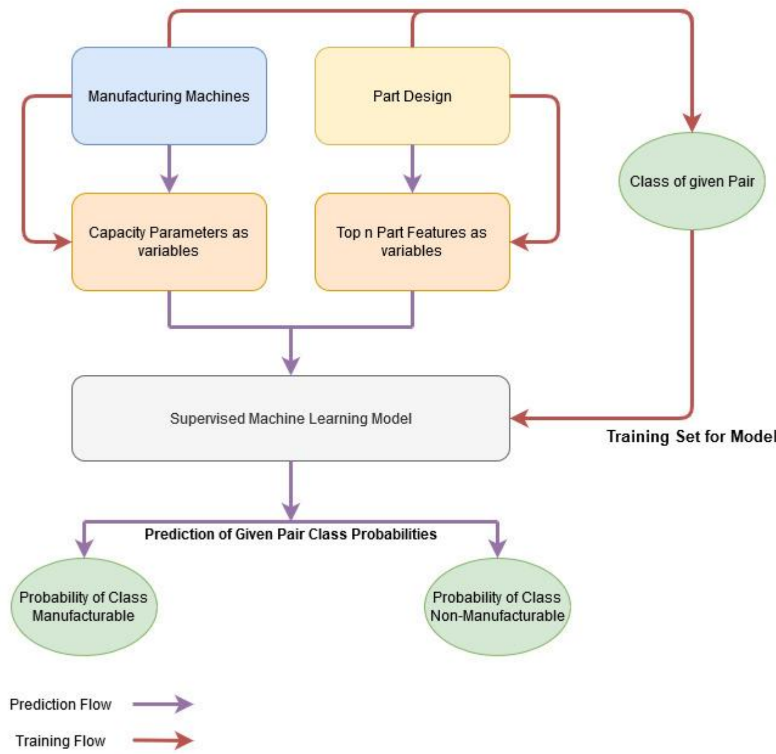

Figure 25. Data Flow in Pattern Matching Section

Table 10. Sample Training Dataset

Variables	Haas VF 3	Haas VF 2-TR	Haas Mini Mill	Doosan Puma 4100L	Haas ST 35
Axis Travel X	40	30	16	9.4	9.4
Axis Travel Y	20	16	12		
Axis Travel Z	25	20	10	52.5	52.5
Table Length	48	36	28.75		
Table Width	18	14	12		
Number of T-Slots	5	3	3		
Max Weight	3500	3000	500		
Chuck Size				18	12
Max Cutting Diameter				21.7	15
Max Cutting Length				82.36	32.5
Max Spindle Nose to Table Length	4.2	4	4		
Min Spindle Nose to Table Length	29.2	24	14		
Spindle Bore Diameter				5.2	4.62
Spindle Nose Type				BMT-KEY	A2-8
Number of Tools Holding Capacity				12	12
Tooling Type	Carousel	SMTC	Carousel		
Capacity	20	30+1	10		
Max Tool Diameter	3.5	2.5	3.5		
Max Tool Weight	12	12	12		
Tool-to-Tool	4.2	2.8	4.2		
Chip-to-Chip	4.5	3.6	5		
Max Cutting	650	650	500		
Max Rating	30	30	7.5	50	40
Max Speed	8100	8100	6000	2000	3200
Max Torque	90	90	33	2419	425
Lubrication	Air/Oil Injection	Air/Oil Injection	Greased Packed		
Cooling	Liquid	Liquid	Air Cooled	Liquid	Liquid
Feature 1_Dimension 1	28.91	28.91	28.91	28.91	28.91
Feature 1_Dimension 2	1.99	1.99	1.99	1.99	1.99
Feature 1_Dimension 3	28.91	28.91	28.91	28.91	28.91
Feature 1_Dimension 4	1.99	1.99	1.99	1.99	1.99
Feature 2_Dimension 1	10	10	10	10	10
Feature 2_Dimension 2	10	10	10	10	10
Feature 2_Dimension 3	10	10	10	10	10
Feature 2_Dimension 4	10	10	10	10	10

Figure 25 shows the schematic data flow for training machine learning models. A supervised machine learning model can be trained using the labels for each pair being either machinable or non-machinable. Upon training a model, the prediction will be the class probabilities. These class probabilities are important since further study can be done on the understanding significance of each feature and capacity parameters from a given pair.

Table 10 shows the sample dataset created based on the sample machine shop considered in Step 1 and the connector part design's dimensional data extracted in the previous step. The largest two features of the connector part are considered here for the creating dataset. The connector part is paired with each of the machines and labeled as manufacturable or non-manufacturable.

The cumulative table created using machine capacity data and part feature dimensions help in bridging the connection between the part fitment and the machine's workspace. As seen in chapter 2, the machine workspace constraints are important parameters to understand the fitment of the given part based on the feature dimensions. Moreover, the feature level extraction of dimensions from part design helps in choosing the features and understanding the pattern of each feature and the fitment. Furthermore, the training set with large training samples covering a significant amount of parts can train the classification models to distinguish between different machines. For instance, from table 10, the milling machine does not have data regarding the chuck whereas the lathe machine does not contain any values in table data fields. These missing fields are the pattern of data of different lathe and milling machines. Any observations of similar missing fields help the model for understanding the type of machine. On the other hand, in the training set, the machinable label signifies the part and machine which can perform the machining.

Whereas, non-machinable denotes the non-compatibility of the part and the machine. Such patterns are important for understanding the pattern. Using these patterns, the proposed model gives the probabilities of both classes for each label. This probability is the compatibility measure of a given observation pair. Since the part data is represented in feature-based logic. These probabilities explain the compatibilities of given part features and the machines. In other words, suppose, a given part has two features and one can be machined on a lathe and the other on the milling machine. If the model is predicted using both the features, the probability of each class will be closer. Whereas, if the prediction is done using individual features, then the probabilities of being machinable will be larger compared to the class probability of being non-machinable.

4.3 Discussion on Implementation

The example machines analyzed are of two types, i) lathe and ii) milling machine. Furthermore, different variants of these machines are also sampled. These variants vary in their respective manufacturing capacities. The variance in their capacity restricts the parts to be manufactured. The capacity data is extracted using web scraping from respective websites. This shows the feasibility of collecting the data for a large number of machine samples without physically inspecting the machines. The example connector part considered can be manufactured on a single machine type. More complex parts that require different processes are not considered in the sample. The dimensions extracted and stored in the XML file helps to understand the dimensions of each face. These faces are conventionally considered for defining the manufacturing processes. Creating a cumulative database of part features and the manufacturing machine's capacity parameters gives extensive data for understanding the manufacturability of a part on the given machine.

Feature matching involves training and testing the classification model. For training a classification model, a large number of data is required. Moreover, such a dataset needs to be labeled for the respective class label of machinable and non-machinable. The suggested active learning approach helps in reducing the number of labeled data, but still, the initial sample of the training set needs to be created using several different parts and machine pairs. The class probabilities do not define the processes involved in manufacturing the part but such probabilities help in understanding the compatibility of the machine and the part. The understanding of the probabilities of being machinable with a given machine reduces the pool of non-machinable machines. This helps the large-scale manufactures for scaling down the pool of machine shops with less capacity for machining the part. In this thesis, the sample machine shop and the connector part formed 5 observations as shown in table 10. Such a small number of the training set is not sufficient for training and testing the performance of the classification model.

Chapter 5: Conclusion and Future Study

The proposed model provides the framework and strategies for data extraction and storage of part design and manufacturing capability data. The model helps in designing, creating, and populating a database for data generated within manufacturing industries. Moreover, the proposed model can be treated as the framework for understanding the dimensionality of the data for use in predictive analytics. Currently, the part data are stored in 2d and 3d CAD models. The algorithm expands the storage methods into numerical and machine learnable format. Part designs mainly explore the geometrical relevance of the feature. Considering this the XML file format uses the geometrical-feature logic for storing the part dimensions. This helps in preserving the geometrical information of the parts. Moreover, the method also considers assemblies and provides single document storage for assemblies with parting into the components using XML tags. This has major application potential for complex assembly and part retrieval process. The XML storage of design data can be used for matching similar faces between the different parts. This can give rise to an index for understanding geometric symmetry between parts.

The demand for new products and time to market are the key driving force for large-scale manufactures. These manufacturers often need integrative and advanced vendor search systems for finding new suppliers for their critical requirements. Whereas small-scale machine shop lacks in approachability for large manufacturers for new projects. This framework helps to connect the requirement-capacity-based market search for large-scale manufactures for enhancing supplier portfolio and provides a platform for small-medium scale manufacturers for seeking new projects.

The proposed framework establishes a connection between small-medium scale manufactures and large-scale manufactures. The linking of desired part features developed by large-scale manufactures and small-scale manufacturer's machine capacity is matched using the data extracted from various manufacturing elements of the system. This mapping quantifies the probability of machining of a given part on the available machine of respective machine shops.

Application of virtual copy of machine shops

Virtual copies of machine shops represent the manufacturing capabilities of machine shops. The capability opens a wide range of applications regarding vendor matching and new product development predictions. As given in Chapter 2, the machine specifications directly imply the compatibility of desired part manufacturing. The capability parameters quantify the machine's workspace and operability strength. The workspace mainly consists of tool holding and swinging dimensions, part holding dimensions, and other tooling parameters such as type of tool holder. Moreover, the operability parameters give extensive knowledge about ranges of spindle speed, feed rates, and torque measures. As seen in chapter 2, process parameters such as spindle speed, torque, and feed rates define the machining accuracy. Storing the data of the machine's operability capacity helps in defining the processes which require higher ranges of process parameters.

Application of part dimensional data

The proposed algorithm for dimensional data extraction serves a vast range of applications. Each part which is stored in 3D format can be stored and used into a dimension based numeric XML file. The part dimensions can be used for creating search engines and further storing the designs into machine recognizable format. The 3D CAD format stores the part design in a 3-dimensional model and often these files are not readable by CAPP tools. An XML file with feature-based dimensions gives a dimensional representation of the part data based on the features. The proposed algorithm reads and extracts the dimensions with respect to each face of the given part design. Such strategy preserves the dimensional and face relevance.

Future Study

The proposed model gives a major insight for creating a database for manufacturing data. As discussed in the model, the machine learning approach is important for matching the pattern between part data and the machine's virtual copies. A dedicated database can be created for generating a significant amount of data for training and validating machine learning models for matching the patterns. Additional data of parts such as the bill of materials and standard information of the part can be added in the XML file of the given part. This will add additional dimensions for the part information and can be used for predictive analytics for new product development as well as for improving the existing product's manufacturing process. This thesis undertakes single machine operations, more complex parts with the requirement of different machining processes on different types of machines are not considered. Additional complex parts and the feature-based machining match can be explored.

The proposed data extraction algorithm undertakes simple parts and more complex parts with curved and uneven special surfaces can be explored for extending the proposed algorithm. Moreover, tolerance data is represented in model-based definition (MBD) files. Instead of the 3D CAD model, MBD models can be used for extending the tolerance storages and enriching the part information storage schema. The tolerancing information of part data can be extracted with a certain extension to the proposed algorithm. The MBD file stores such information in a standard way. In STEP standard, MBD files are presented in separate and dedicated standard file format. Using such standard file format, the proposed algorithm can be extended for extracting type and tolerance values from each face and feature of the part. However, for storing the tolerancing information of the part, the XML schema needs to revise. As shown in this thesis, dimensional data is stored in a features-based logical order. By providing an extension for the proposed XML schema, tolerancing information of each feature, and the type of tolerancing can be stored in an XML file.

References

Abiteboul, S., Buneman, P., & Suciu, D. (2014). *Data on the Web: Semistructured Data and XML*.

Al-wswasi, M., & Ivanov, A. (2019). A novel and smart interactive feature recognition system for rotational parts using a STEP file. *International Journal of Advanced Manufacturing Technology*, *104*(1–4), 261–284. https://doi.org/10.1007/s00170-019-03849-1

Alemanni, M., Destefanis, F., & Vezzetti, E. (2011). Model-based definition design in the product lifecycle management scenario. *International Journal of Advanced Manufacturing Technology*, *52*(1–4), 1–14. https://doi.org/10.1007/s00170-010-2699-y

Bijnens, J., & Cheshire, D. (2019). The current state of model based definition. *Computer-Aided Design and Applications*, *16*(2), 308–317. https://doi.org/10.14733/cadaps.2019.308-317

Chollet, F. (2018). Deep Learning with Python. In Toni Arritola, J. Gaines, A. Dragosavljevic´, & T. Taylor (Eds.), *Deep Learning with Python*. Manning Publications Co. https://doi.org/10.1007/978-1-4842-2766-4

Dimitrov, M. (n.d.). *XML Standards for Ontology Exchange*. 1–68.

Doosan. (2020). *Doosan Machine*. https://www.doosanmachinetools.us

Fang, F. Z., Li, Z., Arokiam, A., & Gorman, T. (2016). Closed Loop PMI Driven Dimensional Quality Lifecycle Management Approach for Smart Manufacturing System. *Procedia CIRP*, *56*, 614–619. https://doi.org/10.1016/j.procir.2016.10.121

Gao, J., Zheng, D., & Gindy, N. (2004). Mathematical representation of feature conversion for CAD/CAM system integration. *Robotics and Computer-Integrated Manufacturing*, *20*(5), 457–467. https://doi.org/10.1016/j.rcim.2004.05.001

Ghorpade, B., Raman, S., Marquez, C., Ramesh, A., & Alam, T. (2020a). *Integrative Framework for Advanced Cyber-Physical Manufacturing Procurement Systems Volume - I*.

Ghorpade, B., Raman, S., Marquez, C., Ramesh, A., & Alam, T. (2020b). *Integrative Framework for Advanced Cyber-Physical Manufacturing Procurement Systems Volume - II*.

Gil, Y., & Ratnakar, V. (2002). A Comparison of (Semantic) Markup Languages. *Proceedings of the Fifteenth International Florida Artificial Intelligence Research Society Conference*, 413–418.

HAAS. (2020). *HAAS Machines*.

Hamdan, A., Sarhan, A. A. D., & Hamdi, M. (2012). An optimization method of the machining parameters in high-speed machining of stainless steel using coated carbide tool for best surface finish. *International Journal of Advanced Manufacturing Technology*, *58*(1–4), 81–91. https://doi.org/10.1007/s00170-011-3392-5

Hirsch-Kreinsen, H., Kubach, U., Stark, R., Wichert, G. von, Hornung, S., Lisa Hubrecht, Sedlmeir, J., & Steglich, S. (2019). Key themes of Industrie 4.0. Research and development needs for successfull implementation of Industrie 4.0 - acatech - National Academy of Science and Engineering. *Reseach Council of the Platform Industrie 4.0*. https://en.acatech.de/publication/key-themes-of-industrie-4-0/

Industrial automation systems and integration product data representation and exchange—part 203. (1994). ISO 10303-203:1994; application protocol: configuration-controlled

3D designs of Mechanical parts and assemblies. *International Organization for Standardization*, *1994*.

Ismail, N., Tan, C. F., Wong, S. V, Osman, M. R., & Sulaiman, S. (2002). *Ruled-Based Feature Extraction and Recognition*. 3–6.

Jasiewicz, M., Miądlicki, K., & Powałka, B. (2018). Assistance of machining parameters selection for slender tools in CNC control. *AIP Conference Proceedings*, *2029*(October). https://doi.org/10.1063/1.5066486

Kalpakjian, S., & Steven, S. (2014). *Manufacturing engineering and technology / Serope Kalpakjian, Steven R. Schmid.* (Seventh). Pearson Education, Inc.

Kathe, C. (2018). *IMPROVING PRODUCT DESIGN TOLERANCES USING METR-ONTOLOGY* (Vol. 2). University of Oklahoma.

Kremer, J., Steenstrup Pedersen, K., & Igel, C. (2014). Active learning with support vector machines. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *4*(4), 313–326. https://doi.org/10.1002/widm.1132

Kuhn, M., & Johnson, K. (2013). Applied predictive modeling. In *Applied Predictive Modeling*. https://doi.org/10.1007/978-1-4614-6849-3

Kutin, A. A., Bushuev, V. V., & Molodtsov, V. V. (2019). Digital twins of mechatronic machine tools for modern manufacturing. *IOP Conference Series: Materials Science and Engineering*, *568*(1). https://doi.org/10.1088/1757-899X/568/1/012070

Li, J., Wang, P., Chen, X. C., & Yang, T. J. (2016). A study on the optimal selection of spur slice cutter parameters and machining parameters. *International Journal of Advanced Manufacturing Technology*, *82*(1–4), 407–417. https://doi.org/10.1007/s00170-015-7369-7

Liu, W., Yang, J., Li, P., Han, Y., Zhao, J., & Shi, H. (2018). A novel object-based supervised classification method with active learning and random forest for PolSAR imagery. *Remote Sensing*, *10*(7). https://doi.org/10.3390/rs10071092

Liu, Y. (2004). Active learning with support vector machine applied to gene expression data for cancer classification. *Journal of Chemical Information and Computer Sciences*, *44*(6), 1936–1941. https://doi.org/10.1021/ci049810a

MALLESWARI, V. N. (2013). *Automatic Feature Recognition for* [ANDHRA UNIVERSITY COLLEGE OF ENGINEERING]. http://hdl.handle.net/10603/14116

Manna, A., & Bhattacharayya, B. (2005). Influence of machining parameters on the machinability of particulate reinforced Al/SiC-MMC. *International Journal of Advanced Manufacturing Technology*, *25*(9–10), 850–856. https://doi.org/10.1007/s00170-003-1917-2

Mazak. (2020). *Mazak Machine*. https://www.mazakusa.com/?gclid=Cj0KCQiAnb79BRDgARIsAOVbhRqXf5A9b mU64YNn8vvd1h3wNrHA QiOXh7mBVvkE7KZRrES2_YUlgcoaAoWTEALw_wcB

Muruganandam, S., & Pugazhenthi, S. (2010). Selection of optimal machining parameters for hexapod machine tool. *International Journal of Advanced Manufacturing Technology*, *46*(5–8), 801–810. https://doi.org/10.1007/s00170-009-2123-7

Patil, L., Dutta, D., & Sriram, R. (2005). Ontology formalization of product semantics for product lifecycle management. *Proceedings of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference - DETC2005*, *3 B*(May 2014), 809–816.

https://doi.org/10.1115/detc2005-85121

Pawar, P. J., & Rao, R. V. (2013). Parameter optimization of machining processes using teaching-learning-based optimization algorithm. *International Journal of Advanced Manufacturing Technology*, *67*(5–8), 995–1006. https://doi.org/10.1007/s00170-012-4524-2

Sateesh, P., & P, M. (2017). Subject : Mechanical Engineering IJRIME A METHODOLOGY FOR FEATURE EXTRACTION AND RECOGNITION FOR CAD / CAM INTEGRATION USING STEP FILE ™© all copyrights reserved. *INTERNATIONAL JOURNAL OF RESEARCH AND INNOVATION Subject:*, *4*(1), 711–725.

Schools of advanced study university of London. (2020). *An introduction to markup*. https://port.sas.ac.uk/mod/book/view.php?id=568&chapterid=336

ScienceDirect. (2020). *Eucledian Distance*. https://www.sciencedirect.com/topics/mathematics/euclidean-distance

Settles, B. (2010). Active Learning Literature Survey. In *Materials Letters*. University ofWisconsin–Madison.

Silberschatz, A., Korth, H. F., & Sudarshan, S. (2011). *DATABASE SYSTEM CONCEPTS* (Sixth). McGraw Hill Companies, Inc.

Sormaz, D., Gouveia, R., & Sarkar, A. (2018). Rule Based Process Selection of Milling Processes Based on Gd&T Requirements. *Journal of Production Engineering*, *21*(2), 19–26. https://doi.org/10.24867/jpe-2018-02-019

Theissen, N., Laspas, T., Szipka, K., & Archenti, A. (2018). Virtual machining system simulator: Analysis of machine tool accuracy. *Procedia Manufacturing*, *25*, 338–343.

https://doi.org/10.1016/j.promfg.2018.06.101

Van Ruijven, L. C. (2013). Ontology for systems engineering. *Procedia Computer Science*, *16*, 383–392. https://doi.org/10.1016/j.procs.2013.01.040

Verhagen, W. J. C. (2015). Concurrent Engineering in the 21st Century. In *Concurrent Engineering in the 21st Century*. https://doi.org/10.1007/978-3-319-13776-6

Waurzyniak, P. (2015). *Why Manufacturing Needs Real-Time Data Collection*. SME. https://www.sme.org/technologies/articles/2015/october/manufacturing-needs-real-time-data-collection/

Yusup, N., Zain, A. M., & Hashim, S. Z. M. (2012). Evolutionary techniques in optimizing machining parameters: Review and recent applications (2007-2011). *Expert Systems with Applications*, *39*(10), 9909–9927. https://doi.org/10.1016/j.eswa.2012.02.109

Zainal, N., Zain, A. M., Radzi, N. H. M., & Othman, M. R. (2016). Glowworm swarm optimization (GSO) for optimization of machining parameters. *Journal of Intelligent Manufacturing*, *27*(4), 797–804. https://doi.org/10.1007/s10845-014-0914-7

Appendix

Appendix a. XML file for Connector part dimensional data

As explained in section 4.2.4


```
  <?xml version="1.0"?>
- <Component_1>
  - <Feature_315>
      <Dimensions_Circular Radius="2.400000000000000355"/>
      <Dimensions_Linear Length="3.000000000000001"/>
      <Dimensions_Circular Radius="2.400000000000000355"/>
      <Dimensions_Linear Length="3.000000000000001"/>
    </Feature_315>
  - <Feature_209>
      <Dimensions_Circular Radius="6.000000000000000888"/>
      <Dimensions_Linear Length="1.0000000000000018"/>
      <Dimensions_Linear Length="1.9999999999999991"/>
      <Dimensions_Circular Radius="6.000000000000000888"/>
      <Dimensions_Linear Length="1.9999999999999991"/>
      <Dimensions_Linear Length="1.0000000000000018"/>
    </Feature_209>
  - <Feature_93>
      <Dimensions_Circular Radius="5.000000000000000000"/>
      <Dimensions_Linear Length="10.0"/>
      <Dimensions_Circular Radius="5.000000000000000000"/>
      <Dimensions_Linear Length="10.0"/>
    </Feature_93>
  - <Feature_191>
      <Dimensions_Linear Length="1.9999999999999991"/>
      <Dimensions_Circular Radius="10.00000000000000000"/>
      <Dimensions_Linear Length="1.9999999999999991"/>
      <Dimensions_Circular Radius="10.00000000000000000"/>
    </Feature_191>
  - <Feature_239>
      <Dimensions_Linear Length="10.0"/>
      <Dimensions_Circular Radius="10.00000000000000000"/>
      <Dimensions_Linear Length="10.0"/>
      <Dimensions_Circular Radius="10.00000000000000000"/>
    </Feature_239>
```

- <Feature_43>
<Dimensions_Circular Radius="5.0000000000000000000"/>
<Dimensions_Circular Radius="5.0000000000000000000"/>
</Feature_43>
- <Feature_623>
<Dimensions_Circular Radius="5.0000000000000000000"/>
<Dimensions_Circular Radius="5.0000000000000000000"/>
</Feature_623>
- <Feature_548>
<Dimensions_Linear Length="10.0"/>
<Dimensions_Circular Radius="5.0000000000000000000"/>
<Dimensions_Linear Length="10.0"/>
<Dimensions_Circular Radius="5.0000000000000000000"/>
</Feature_548>
- <Feature_563>
<Dimensions_Linear Length="28.913664589601918"/>
<Dimensions_Linear Length="1.9999999999999991"/>
<Dimensions_Linear Length="28.91366458960192"/>
<Dimensions_Linear Length="1.9999999999999991"/>
</Feature_563>
- <Feature_277>
<Dimensions_Linear Length="28.91366458960192"/>
<Dimensions_Linear Length="1.9999999999999991"/>
<Dimensions_Linear Length="28.91366458960192"/>
<Dimensions_Linear Length="1.9999999999999991"/>
</Feature_277>
- <Feature_611>
<Dimensions_Circular Radius="6.0000000000000000888"/>
<Dimensions_Linear Length="28.91366458960192"/>
<Dimensions_Circular Radius="10.000000000000000000"/>
<Dimensions_Linear Length="28.91366458960192"/>
</Feature_611>
- <Feature_64>
<Dimensions_Circular Radius="2.4000000000000000355"/>
<Dimensions_Circular Radius="2.4000000000000000355"/>
</Feature_64>

- <Feature_664>
<Dimensions_Linear Length="1.0000000000000018"/>
<Dimensions_Circular Radius="6.0000000000000000888"/>
<Dimensions_Linear Length="1.0000000000000018"/>
<Dimensions_Circular Radius="6.0000000000000000888"/>
</Feature_664>
- <Feature_546>
<Dimensions_Circular Radius="2.4000000000000000355"/>
<Dimensions_Circular Radius="2.4000000000000000355"/>
</Feature_546>
- <Feature_151>
<Dimensions_Linear Length="3.000000000000001"/>
<Dimensions_Circular Radius="2.4000000000000000355"/>
<Dimensions_Linear Length="3.000000000000001"/>
<Dimensions_Circular Radius="2.4000000000000000355"/>
</Feature_151>
- <Feature_7>
<Dimensions_Linear Length="9.774663829417616"/>
<Dimensions_Linear Length="6.0"/>
<Dimensions_Linear Length="9.77466382941761"/>
<Dimensions_Linear Length="6.0"/>
</Feature_7>
- <Feature_472>
<Dimensions_Linear Length="8.0"/>
<Dimensions_Linear Length="6.0"/>
<Dimensions_Linear Length="8.0"/>
<Dimensions_Linear Length="6.0"/>
</Feature_472>
- <Feature_202>
<Dimensions_Linear Length="9.774663829417625"/>
<Dimensions_Linear Length="6.0"/>
<Dimensions_Linear Length="9.774663829417623"/>
<Dimensions_Linear Length="6.0"/>
</Feature_202>

- <Feature_13>
<Dimensions_Linear Length="9.77466382941761"/>
<Dimensions_Linear Length="8.0"/>
<Dimensions_Linear Length="9.774663829417623"/>
<Dimensions_Circular Radius="10.000000000000000"/>
</Feature_13>
- <Feature_357>
<Dimensions_Linear Length="9.774663829417616"/>
<Dimensions_Circular Radius="10.000000000000000"/>
<Dimensions_Linear Length="9.774663829417625"/>
<Dimensions_Linear Length="8.0"/>
</Feature_357>
</Component_1>