# EXTRACTION OF FORMAL MANUFACTURING RULES FROM UNSTRUCTURED ENGLISH TEXT

BY

SUNGKU KANG

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Mechanical Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2017

Urbana, Illinois

Doctoral Committee:

Professor Debasish Dutta, Chair
Professor Placid Ferreira
Professor Harrison Hyung Min Kim
Dr. Lalit Patil
Dr. Arvind Rangarajan, GE Global Research

ABSTRACT

Semantics-based approaches—founded on the idea of explicitly encoding meaning separately from the data or the application code—are being applied to manufacturing, for example, to enable early manufacturability feedback. These approaches rely on formal, i.e., computer-interpretable, knowledge and rules along with the context or semantics. On the other hand, manufacturing knowledge has been maintained primarily in the form of unstructured English text. It is considered impractical for engineers to author accurate, formal, and structured manufacturing rules. Previous efforts on extracting semantics from unstructured text in manufacturing have focused exclusively on basic concept names and hierarchies. In this context, this dissertation focuses on the development of a semantics-based framework for acquiring more complex manufacturing knowledge, primarily rules, in a formal form, from unstructured English text such as those written in manufacturing handbooks.

This dissertation includes the following specific research tasks. First, it studies the problem in manufacturing domain, proposes the formal rule extraction framework, and demonstrates its feasibility. Second, it extends the framework to complement standard Natural Language Processing (NLP) techniques with manufacturing domain knowledge to resolve ambiguities, called as domain-specific ambiguities, that are due to manufacturing-specific meanings implicit in the English text. Finally, this dissertation extends the framework to identify the cases that need input text validation, and provide the relevant feedback to the user to modify the input text for the extraction of correct rules.

This research also demonstrates the extensibility of the framework. Specifically, the framework was initially developed using the subset of a manufacturing handbook only including milling, metal stamping, and die-casting sections, and then applied to the rest of the manufacturing processes including 30 sections in forming, machining, casting, molding, assembling, and finishing chapters in the book. Case studies are performed to demonstrate the feasibility of the framework on the dataset of 133 sentences. First, the feasibility of the rule extraction framework is shown by extracting correct rules from approx. 57% of the sentences. Second, the effectiveness of ambiguity resolution by complementing standard NLP techniques with manufacturing domain knowledge is demonstrated by an increasing the correct rules to 70%. Lastly, for the remaining 30% of the cases that need input text validation, relevant feedback is provided to the user to modify the input text for the extraction of the correct rules.

It is expected that this research will facilitate the development of formal manufacturing knowledge including complex manufacturing rules. It will thus address an important barrier that has prevented a larger scale application and the adoption of semantic technologies in the field of manufacturing, especially for semantics-based manufacturability analysis.

*To My Father, Mother, Brother, and Fiance.*

# ACKNOWLEDGMENTS

This project would not have been possible without the support of many people. Many thanks to my advisers, Professor Debasish Dutta and Doctor Lalit Patil for guiding me to the right direction. Also thanks to my committee members, Professor Placid Ferreira, Professor Harrison Kim, Doctor Arvind Rangarajan for providing me valuable feedbacks to complete the Preliminary exam and Final defense. Thanks to the people in GE Global Research, Dean Robinson, Arvind Rangarajan, Abha Moitra, Michael Graham, Tao Jia for valuable discussions and thanks to Christine Furstoss, Global Technology Director of Materials and Manufacturing Technologies at GE Global Research, for supporting this work. Thanks to Professor Farhad Ameri, who provided comments about my research and manuscripts. And finally, thanks to my family, fiance, and friends who have endured this long process with me and always offered support and love.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

## LIST OF ABBREVIATIONS

| | |
|---|---|
| AI | Artificial Intelligence |
| ANN | Artificial Neural Network |
| BSA | Basic Static code Analysis |
| CAD | Computer-Aided Design |
| CAPI | Computer-Aided Process Innovation |
| CBM | Constraint-Based Modeling |
| CRF | Conditional Random Fields |
| DFM | Design for Manufacturing/Manufacturability |
| ERD | Entity-Relationship Diagram |
| HPSG | Head-driven Phrase Structure Grammar |
| IBD | Intention-Based Diagnosis |
| ITS | Intelligent Tutoring System |
| OMIM | Online Mendelian Inheritance in Man |
| OWL | Web Ontology Language |
| POS | Part-Of-Speech |
| PT | Program Transformation |
| LDA | Latent Dirichlet Allocation |
| MSDL | Manufacturing Service Description Language |
| MT | Model Tracing |
| NER | Named Entity Recognition |
| NLP | Natural Language Processing |
| RDF | Resource Description Framework |

| | |
|---|---|
| RDFS | Resource Description Framework Scheme |
| SADL | Semantic Application Design Language |
| SAO | Subject-Action-Object |
| SKOS | Simple Knowledge Organization System |
| SVD | Singular Value Decomposition |
| XRML | eXtensive Rule Markup Language |

# CHAPTER 1: INTRODUCTION

## 1.1 Background and Motivation

Semantics-based approaches—relying on explicitly encoded formal knowledge in problem-solving—have been proposed as powerful mechanisms in manufacturing field. The advantage of semantics-based approaches lies in enabling the reusable representation (e.g. ontology) of product information, enterprise knowledge, and manufacturing resources so that better decision-making is possible through the regular access to experts' knowledge [1]. Previous applications of semantics-based approaches in manufacturing field include creating an upper ontology for interoperable manufacturing service description [2], enabling data interoperability between Computer-Aided Design (CAD) systems [3], [4], supporting web service for factory automation process [5], and modeling the tasks for cloud manufacturing [6].

In addition to these applications, Rangarajan et al. [7] demonstrated semantics-based early manufacturability analysis that utilizes formal manufacturing rules to overcome the limitation of traditional Design for Manufacturing (DFM) systems. Specifically, as semantics-based manufacturability analysis relies on formal manufacturing knowledge decoupled from a certain type of CAD environment, it provides the following advantages [7]: 1) It can be easily adapted to different types of CAD environments. 2) The formal manufacturing rules can be easily modified and extended. 3) It is expected to enable natural language-based manufacturing rule authoring for manufacturing and design engineers. 4) It is also expected to support logical reasoning to get design suggestions for what-if scenario.

However, even though manufacturers seem to accept the above-mentioned value that semantics-based approaches can provide, the adoption rate of the approaches in manufacturing industry is still slow. One of the fundamental bottlenecks in adopting semantics-based approaches in manufacturing is the lack of formal knowledge acquisition mechanism. Specifically, while the most prevalent form of encoding manufacturing knowledge is unstructured text (approximately 80% [8], [9]), extracting complex manufacturing knowledge from the text has been elusive. Therefore, it is expected that engineers would author formal manufacturing rules in formal structured languages such as the Web Ontology Language (OWL) [10] or Semantic Application Design Language (SADL) [11]. However, this assumption is impractical since it requires engineers, whose main job is to design and manufacture, to learn the formal structured languages and translate the text into formal manufacturing knowledge. This makes the semantics-based analysis less feasible and reliable. In this context, the primary challenge that motivates this dissertation is the lack of a reliable (semi-)automated method to extract formal manufacturing knowledge, especially complex formal manufacturing rules, from unstructured text such as those written in a design for manufacturing handbook.

In the context of the above-mentioned challenge, the rest of this chapter is organized as follows. First, the overall goal of this dissertation is presented in section 1.2. Then the specific research tasks, scope, and challenges of this dissertation are presented from section 1.3 to section 1.5. Lastly, the outline

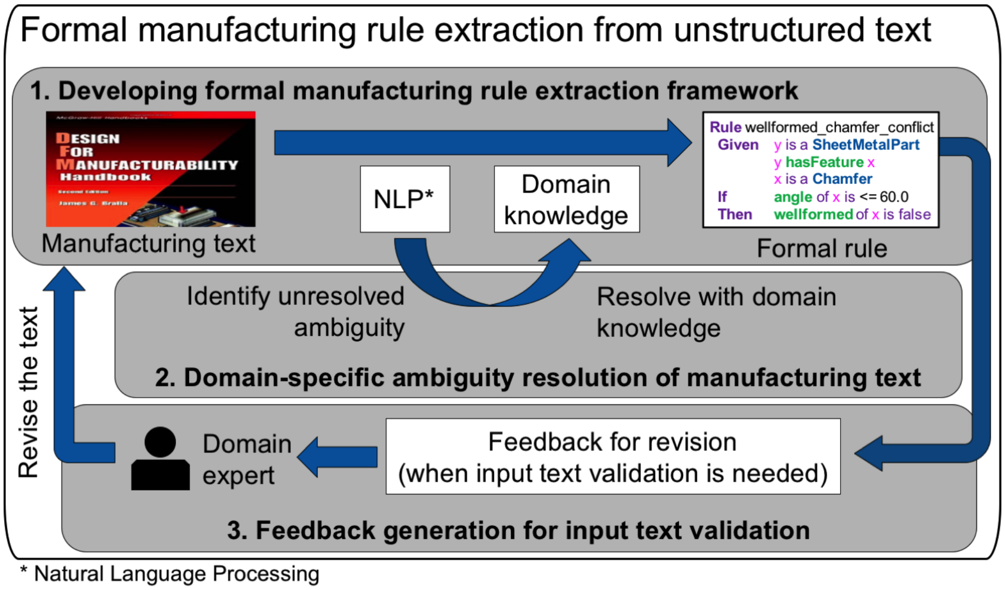



Figure 1.1: The overview of this dissertation

of this dissertation is presented in section 1.6.

## 1.2 Dissertation Goal

The overall goal of this dissertation is to address the gap of formal knowledge acquisition issue that has limited the adoption of semantics-based approaches in manufacturing, especially semantics-based manufacturability analysis. Figure 1.1 shows the overall framework of formal manufacturing rule extraction to achieve the goal. From unstructured English manufacturing text with rule-like information (shown at the top-left of Figure 1.1), the framework resolves the ambiguity and extract the formal manufacturing rule (shown at the top-right of Figure 1.1) utilizing Natural Language Processing (NLP) and domain knowledge. For the cases that need input text validation, relevant feedback is provided so that the user can modify the input text to acquire correct rules (shown at the bottom of Figure 1.1).

## 1.3 Research Tasks

Each shaded region of Figure 1.1 represents the specific research task of this dissertation. The description of each research task is as follows:

- Design and develop a framework to extract complex formal manufacturing knowledge, especially manufacturing rules in a semantically-usable form, from unstructured English text such as those written in a manufacturing handbook

- Extends the rule extraction framework to address unresolved ambiguity in manufacturing text, called as domain-specific ambiguity due to domain-dependent meaning implicit in the English text, by complementing standard NLP techniques with domain knowledge

- Extends the rule extraction framework to provide feedback for the cases that need input text validation so that the user can modify the input text to acquire correct rules

## 1.4 Research Scope

In order to limit the scope of the works on formal manufacturing rule extraction, this dissertation considers the design for manufacturing handbook, written by Bralla [12], as the source of the dataset. In addition, the scope of the dataset is incrementally extended to verify the applicability of the framework. Specifically, the rule extraction framework is initially developed using the 3 sections of the handbook, milling, metal stamping, and die-casting, and then extended to 30 sections in forming, machining, casting, molding, assembling, and finishing chapters of the book. In addition, in creating the dataset, this dissertation only considers the following types of sentences: 1) the sentences specifying the constraint between manufacturing features and 2) the sentences specifying the clear preference to a design over another design. For the additional details of the dataset creation and the full list of the sentences in the dataset, the readers are referred to Appendix A of this dissertation. Please note that the goal of this research is not to enable rule extraction for all handbooks, but to develop and demonstrate a framework that would be customizable. This aligns with the industry expectations of a framework that would then be customized as per its internally authored rules.

## 1.5 Research Challenges

This section presents the research challenges for each of the three research tasks.

### 1.5.1 Developing Framework to Extract Formal Manufacturing Rule from Text

The problem of extracting formal manufacturing rule from unstructured English text is challenging for the following reasons:

**Identifying manufacturing concepts and inter-concept relations in text** For formal manufacturing rule extraction, it is important to identify the terms corresponding to manufacturing-related concepts and their relations. It is challenging since it requires the analysis of highly domain-specific unstructured text and the mapping between the text and formal manufacturing knowledge.

**Combining the concepts and relations to form a coherent rule** It is challenging since a formal manufacturing rule should be constructed considering the relations among the manufacturing concepts expressed with different expressions or grammar structures. For example, the sentences with the expressions such as "if . . . then", "should", and "preferred to" should be translated in different ways.

### 1.5.2 Developing Domain-specific Ambiguity Resolution Method for Manufacturing Text

The problem of resolving domain-specific ambiguity of manufacturing text is challenging for the following reasons:

**Identifying domain-specific ambiguity in text** It is important to identify domain-specific ambiguity since it leads to the extraction of undesired or incorrect rules. It is challenging since standard NLP techniques cannot capture the existence of an unresolved ambiguity in a manufacturing text.

**Considering manufacturing domain knowledge to resolve ambiguity** This entails the development of heuristics to consider manufacturing domain knowledge, which is not directly written in the text but present in the manufacturing domain ontology, to resolve ambiguities. It is challenging since standard NLP techniques do not have a mechanism to incorporate manufacturing domain knowledge for ambiguity resolution.

### 1.5.3 Developing Feedback Generation Method for input text validation

The problem of generating the relevant feedback for input text validation is challenging for the following reasons:

**Identifying the cases that need input text validation** It is challenging since it requires the modeling of the cases that need input text validation. The types of the cases include 1) the inclusion of unnecessary information in the extracted rule, 2) the inclusion of semantically invalid expression, and 3) the absence of explicit subject in the input sentence.

**Generating feedback for input text validation** It is challenging since the framework should be able to provide the relevant feedback for the different types of issues. It is also challenging since the feedback should be able to suggest the options that can be performed by the user without learning the formal structured languages.

## 1.6 Dissertation Overview

The remainder of this dissertation is organized as follows. First, Chapter 2 of this dissertation presents the development of the formal manufacturing rule extraction framework that utilizes standard NLP techniques and the manually constructed manufacturing domain ontology. Chapter 3 of this dissertation presents the development of the domain-specific ambiguity resolution method that complements standard NLP techniques with manufacturing domain knowledge. Specifically, the rule extraction framework is extended to adopt the ambiguity resolution method, and the effectiveness of the method is demonstrated by significantly increasing the correct rules. Chapter 4 of this dissertation presents the development of the feedback generation method that utilizes Constraint-based Modeling (CBM) coupled with standard NLP techniques and the manufacturing domain ontology. Specifically, the rule extraction framework is further extended to adopt the feedback generation method, and the effectiveness of the method is demonstrated by enabling the extraction of correct rules from the cases that need input text validation after the user modifies the input text based on the generated feedback. Lastly, Chapter 5 presents the contributions of this dissertation, the possible applications of the presented works, and the future works that can improve the research of formal manufacturing rule extraction.

# CHAPTER 2: FORMAL MANUFACTURING RULE EXTRACTION FRAMEWORK

## 2.1 Motivation

As mentioned in the previous chapter, semantics-based manufacturability analysis is expected to overcome the limitations of traditional manufacturability analysis. However, the lack of (semi-)automated mechanism to acquire formal manufacturing rules from text has limited the application of semantics-based manufacturability analysis. In this chapter, formal manufacturing rule extraction framework is designed and developed to address the issue. In addition, the feasibility of the framework is verified from the dataset created from the design for manufacturing handbook written by Bralla [12].

The remainder of this chapter is organized as follows. First, Section 2.2 reviews the existing works relevant to manufacturability analysis, the extraction of formal manufacturing knowledge from text, and formal rule extraction approaches outside of manufacturing domain. Then Section 2.3 defines the main problem of this chapter, and Section 2.4 presents the overview of the developed framework. The detailed processes of rule extraction are presented in Section 2.5 and Section 2.6. Lastly, the implementation of the framework and experimental result are shown in Section 2.7, and Section 2.8 summarizes this chapter.

## 2.2 Relevant Work

### 2.2.1 Manufacturability Analysis

DFM, which began in the early 1990s, refers to the process or practice of designing products focusing on the manufacturability of the products. For decades, the importance of DFM has been emphasized in design and manufacturing domain, and it becomes an integral part of the product development process [12]. The value of employing DFM lies in decreasing the iterations between design and manufacturing phase [7]. Reducing the iterations provides the following advantages: 1) reduced manufacturing cost by allowing potential problems to be fixed in the design phase, which is the least expensive phase to address the problems and 2) reduced lead time for new product introduction, which is especially critical in the fast-changing industry. Manufacturability analysis has been studied to incorporate important functionalities required by a DFM tool as follows [13]:

- Analyzing a design to decide whether it is manufacturable or not.
- Identifying features that are the impediments for manufacturing.
- providing advice on how to eliminate un-manufacturable features, decrease costs, and increase yield.

In the rest of this section, traditional manufacturability analysis and its limitations are introduced. Then the advantages of semantics-based manufacturability analysis are followed.

#### 2.2.1.1 Traditional Manufacturability Analysis and Its Limitation

There are several commercial DFM tools performing manufacturability analysis including DFMPro [14] for SolidWorks [15], Checkmate plugin for NX [16], the tool from Boothroyd and DewHurst Inc. [17], and Apriori [18]. As these DFM tools rely on the hard-coded rules embedded in the platform to perform manufacturability analysis, they have the following limitations:

- The DFM tools are platform-dependent. Therefore, they are only applicable to the compatible platform.

- Most of the DFM tools provide fixed rule database, typically created by the application developers. Therefore, the tools may not provide the desired manufacturing rules. Moreover, confidentiality issues make it difficult to share the desired manufacturing rule with the application developers to be included in the tools.

- Even though some of the tools (e.g. Checkmate plugin for NX [16]) allow the addition of custom rules, they require the understanding of their feature recognition schemes and/or dedicated programming languages. Therefore, dedicated software programmers may be needed to author rules rather than being done by onsite design or manufacturing engineers.

Other than commercial tools, customized expert systems have been designed for manufacturability analysis using various methods. The most popular methods are rule-based system and object-oriented technique. First, rule-based systems [19]–[26] use IF-THEN clauses with logical combinations to represent their knowledge base using logic programming languages such as LISP and Prolog. The advantage of the methods is that IF-THEN clauses are convenient to model the functionalities for manufacturability analysis. On the other hand, object-oriented techniques [27]–[30] utilize object-oriented models that define objects, which correspond to manufacturing resources or capabilities, with properties and relations between the objects. The advantage of the methods lies in considering the inter-object relations in manufacturability analysis. The combinations of the two methods are also actively employed for manufacturability analysis. The hybrid systems include those of Venkatachalam et al. [31], Jia et al. [32], and Ramana and Rao [33].

The other methods include analytical hierarchy process and neural network-based methods. Analytical hierarchy process is a structured methodology for analyzing complex decisions and finding the best alternative solution. Ong et al. [34], [35] demonstrated the analytical hierarchy process-based manufacturability analysis system that estimates manufacturability index, which is the indicator of the relative ease of manufacturing, based on the different weights of the features relevant to manufacturability. Compared to the other methods, neural network-based method is a relatively recent technique that utilizes an artificial neural network to approximate complex non-linear relations. For example, Korosec et al. [36] utilized the relationship between surface finish, surface hardness, and part manufacturability to train a neural network for a manufacturability analysis system.

However, the expert systems also share the same limitation of the commercial

DFM tools. Specifically, manufacturing rules are embedded in the system using complex logic and/or computer programming languages. Although there are some efforts including that of Zhao and Shah [37] which tried decoupling the rules from the application, the underlying structures are still similar to the existing systems. Therefore, the approaches cannot address the requirements such as a simple rule-authoring framework and a designer feedback system.

#### 2.2.1.2 The Advantage of Semantics-based Manufacturability Analysis

To overcome the limitations of traditional manufacturability analysis, Rangarajan et al. suggested the semantics-based manufacturability analysis framework [7]. Specifically, rather than relying on hard-coded rules, their approach utilizes the explicitly encoded semantic model and formal rules separated from the application to provide the following advantages:

- As a semantic model and formal rules are decoupled from a certain CAD environment, semantics-based manufacturability analysis can be easily adapted to different environments. In addition, the semantic model can be easily extended or modified separately from the environment.

- As semantic model focuses on the meaning of domain concepts and relations, semantics-based manufacturability analysis enables the use of ordinary English or English-like languages in the entire process. Therefore, semantics-based approaches do not require design and manufacturing engineers to master complex feature recognition schemes or programming languages for manufacturing rule authoring and management.

- Using the semantic and logical relations defined in the semantic model, computer reasoning engines can be used to infer relations between the asserted models without the intervention of a domain expert.

- Engineers can query the semantic model and obtain quantifiable suggestions. For example, an engineer can query that if a change of design, such as changing material, leads to the violation of any manufacturability rule.

On the other hand, industrial manufacturing knowledge is primarily maintained in documents such as handbooks and guidelines. Therefore, the adoption of the semantics-based approach requires the development of formal manufacturing knowledge from manufacturing-related documents. In the next section, the works relevant to the extraction of manufacturing knowledge from unstructured text are reviewed.

### 2.2.2 Extraction of Manufacturing Knowledge from Text

A considerable amount of manufacturing knowledge is maintained in text (approximately 80% in industry [8], [9]). Since the reuse of textual knowledge has been difficult and time-consuming [7], [38], (semi-)automatically acquiring useful information from unstructured text has been actively studied in manufacturing domain. Various approaches have been tried to extract formal manufacturing knowledge from unstructured text.

First, the efforts in manufacturing domain include the automatic classification of documents. Riel et al. [39] demonstrated the classification of conference papers using the tool named Content Analysis Toolkit (CAT) by InduTech [40]. Boonyasopon et al. [41] improved the work of Riel et al. [39] using Latent Dirichlet Allocation (LDA) and Wikify [42], which is the tool to link terms to the corresponding Wikipedia articles. Similarly, Shotorbani et al. [43] used the combination of LDA and K-means for document clustering and topic modeling. Yazdizadeh et al. [44] utilized a Naïve Bayes classifier to classify the category of manufacturing suppliers. Kaijun et al. [45] demonstrated document classification using genetic algorithm.

The extraction of structured knowledge from text has also been actively studied. Li et al. [46] utilized NLP techniques to extract the structured and semantics-based representation of a design document that aligns with the predefined ontology. Yang et al. [47] demonstrated the automated creation of a thesaurus using Singular Value Decomposition (SVD) and showed that the thesaurus can improve the retrieval of design information from documents. Li et al. [48] extracted design concepts and relations from linguistic patterns of a document and used a domain ontology to create the structured concept graph of design information. Choi et al. [49] demonstrated the extraction of technology tree diagram (TechTree) from Subject-Action-Object (SAO) structures categorized with WordNet-based similarity measure between sentences. Similarly, Cheong et al. also utilized SAO structures from text to acquire functional [50] and system structure knowledge [51] using Functional Basis terms [52], WordNet [53], and word2vec [54]. Lan et al. [55] demonstrated the extraction of design information using the combination of content-based document clustering, named entity recognition (NER) and frequency-based entity relationship detection. Wang et al. [56] extracted process contradiction matrices from patent documents using NLP and a domain ontology to support Computer-Aided Process Innovation (CAPI). Jeon et al. [57] demonstrated the retrieval of CAD models from design documents. Specifically, they first created the semantic representations of design documents and CAD models based on a domain ontology. Then they measured the similarity between the semantic representations to retrieve the most relevant CAD model.

The acquisition and maintenance of ontology have been also studied. Ahmed et al. [58] proposed a methodology for creating an engineering design ontology mainly relying on the interview with domain experts. Li et al. [59], [60] demonstrated a semi-automated methodology incorporating NLP for developing engineering ontologies for indexing unstructured engineering documents and facilitating design information retrieval. Ameri et al. [61] demonstrated the methodology of creating a controlled vocabulary from text and ontological conceptualization based on the controlled vocabulary.

Nevertheless, to the best of knowledge, the extraction of complex manufacturing rules from unstructured text has not received much attention. The following subsection introduces the works relevant to formal rule extraction in various domains.

### 2.2.3 Formal Rule Extraction

In this section, the previous efforts on the extraction of formal rule from unstructured text are reviewed. It should be noted that the focusing is on the extraction of complex rules like business rule, regulation, manufacturing rule, which are formed from multiple relations between concepts to specify the desired behavior for a given condition. In other words, the extraction of simple association rules, such as “A is B” or “A has B” that defines the hierarchical or inclusion relation between two concepts, is not the focus of the literature review in this section. In addition to extracting the relations between concepts, the extraction of a complex rule requires modeling multiple relations to obtain the desired rule [62].

A few efforts have focused on the extraction of complex formal rules from a text using a mark-up language. For example, Kang et al. [62] proposed the use of eXtensive Rule Markup Language (XRML) [63] to help a user mark implicit rules in a web page and identify the components of such rules. However, their approach seems to require extensive user interaction with a large amount of text and not widely accepted.

Most of the later approaches rely on the domain ontology related to the application domain. The approaches including those of Park et al. [64], Singh et al. [65], and Santhakumar et al. [66]. All of them utilized a domain ontology to extract rules from websites of simple domains such as online shopping and car rental. Specifically, they created a domain ontology based on several sample websites and utilized the domain ontology along with search algorithms to extract rules from the different websites of the same domain. The limitation of their approaches is that the performance highly depends on the similarity of the websites used for the domain ontology creation and rule extraction. Hassanpour et al. [67] demonstrated a more systematic framework to extract rules from online text. They used NLP to analyze the Internet text and utilized a domain ontology that contains pre-defined rules in extracting rules from web pages. However, the approach is less reliable when the target domain is complex since it utilizes the pre-defined rules as the templates to create new rules. Similarly, Ye [68], [69] defined the templates of semantic service rules and utilized WordNet [53] to map textual service rules into the semantic rule conforming to one of the templates, but the expressiveness of the extracted rules is limited since the approach also relies on a small number of the pre-defined templates.

Machine learning-based approaches are also actively studied especially in medical informatics, due to the availability of standard corpus relevant to medical domain. For example, Manine et al. [70] annotated Medical corpus (LLL05 corpus [71]) with their rule ontology and utilized machine learning to extract genic interaction rules from text. Hou et al. [72] utilized the annotated corpus Online Mendelian Inheritance in Man (OMIM) [73] and used inductive logic programming [74] as the machine learning framework to derive gene-disease relationships. Thesaurus such as WordNet [53] is also used to support rule extraction. For example, Boufrida et al. [75] utilized NLP to analyze the text and derived the semantic similarity between terms using WordNet [53]. They used the semantic similarity to map the terms into domain-specific formal rules. While these approaches utilized powerful machine learning technique, it is not applicable to manufacturing domain due to the lack of standard, preferably annotated, corpus to train a classifier.

### 2.2.4 Review of the Relevant Works

To the best of knowledge, the extraction of complex manufacturing rules from unstructured text has not been studied. Moreover, as mentioned, the previous approaches for other domains do not fit for the extraction of formal manufacturing rule. This lack of the rule extraction method in manufacturing field is regarded as one of the fundamental gaps preventing the adoption of semantics-based approach in manufacturing industry, and that is the reason this dissertation focuses on the acquisition of formal manufacturing rules from unstructured manufacturing text such as those written in manufacturing handbooks.

To address the challenge, this chapter demonstrates the formal manufacturing rule extraction framework that utilizes standard NLP techniques along with a manufacturing domain ontology. Specifically, the framework analyzes manufacturing text with standard NLP techniques and transforms the NLP results to manufacturing concepts/relations using a controlled vocabulary. Then the manufacturing concepts and relations are utilized to construct the coherent formal manufacturing rule based on a formal manufacturing rule model.

## 2.3 Problem Definition

Figure 2.1 shows the motivating example of formal manufacturing rule extraction framework. The input of the framework, shown at the top of Figure 2.1, is a manufacturing text with rule-like information. Specifically, the input text describes the preferable condition for good manufacturability, which is the perpendicularity between the entrance surface and axis of a hole. The main function of the framework is automatically extracting the equivalent formal manufacturing rule, which is the desired output shown at the bottom of Figure 2.1. Specifically, the output rule is written to identify if the entrance surface and axis are not perpendicular each other, as the objective of manufacturability analysis is to identify the impediment of manufacturing. The output rule also defines the feedback provided when the rule is triggered. Specifically, when the rule is triggered, a negative feedback and the input text are provided as the comment so that the user can address the manufacturabilty issue accordingly. In this dissertation, Semantic Application Design Language (SADL) [11] is used to encode a formal manufacturing rule. A SADL rule [11] consists of formal statements (i.e., triples) created by assembling instances, formal classes, and formal properties. Figure 2.1 shows that instances (magenta) correspond to the manufacturing concepts related to the rule (e.g. *entrance_surface* or *hole_axis*), formal classes (blue) define the types of the instances (e.g. *EntranceSurface* or *HoleAxis*), and formal properties (green) define the logical or semantic relations between the instances (e.g. *hasHoleAxis* or *isPerpendicularTo*). Developing the desired framework requires addressing the following requirements.

**Ability to identify manufacturing concepts in text** It is important to identify the terms corresponding to manufacturing-related concepts in the text and map them to the formally defined manufacturing knowledge because they form the desired formal manufacturing rule. For instance, the motivating example (Figure 2.1) shows that manufacturing-related terms such as "entrance surface" and "hole" are identified and mapped to the formal manufacturing

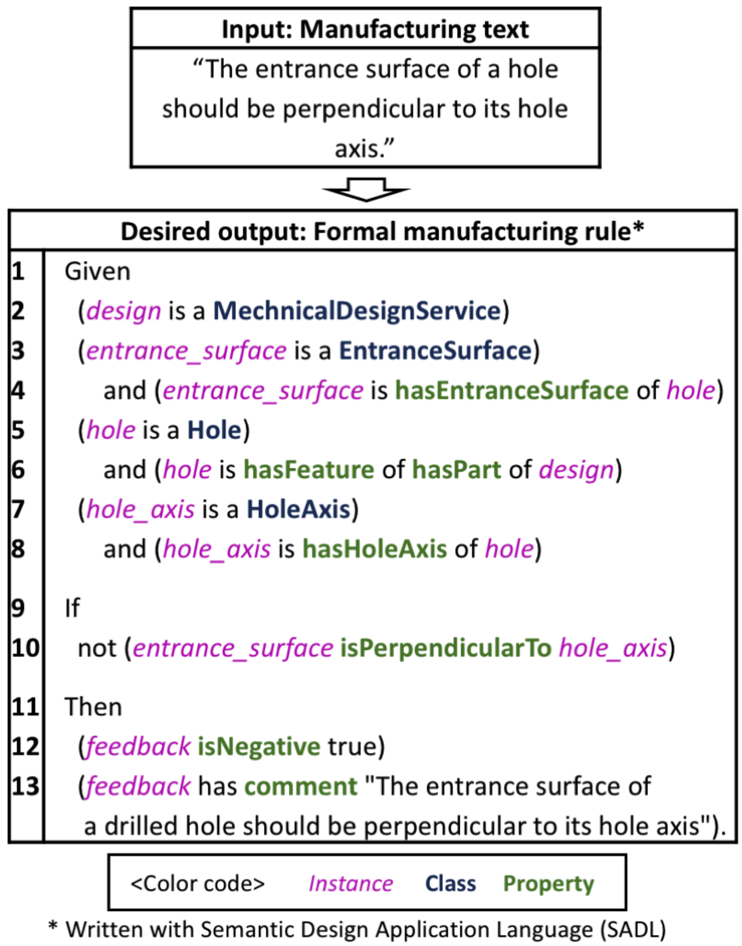



Figure 2.1: The motivating example of formal manufacturing rule extraction framework is shown. The input text (top) describes the preferable condition for good manufacturability, and the desired output rule (bottom) is written to detect if the preferable condition is violated so that the manufacturability issue can be addressed. In this dissertation, we use Semantic Application Design Language (SADL) [11] to encode formal manufacturing rules.

concepts *EntranceSurface* and *Hole* respectively.

**Ability to identify inter-concept relations in text** This is important because the detailed semantics of a formal manufacturing rule is expressed using the inter-concept relations between manufacturing concepts. For instance, the motivating example (Figure 2.1) shows that the inter-concept relation *isPerpendicularTo* is developed from the phrase "be perpendicular to" to represent the perpendicularity between the "entrance surface" and the "hole axis". It should be noted that implicit relations should also be captured explicitly. The examples include the ownership of the hole axis, which is developed from the phrase "its hole axis" (shown at line 8 of Figure 2.1), and the ownership of the entrance surface, which is developed from the phrase "the entrance surface of a hole" (shown at line 4 of Figure 2.1).

**Ability to combine the relations to form a coherent rule** This is important because the relations among the manufacturing concepts are expressed using different expressions or grammar structures (e.g., "if ... then", "should", and "preferred to") from which a coherent and valid formal rule should be extracted. For instance, the motivating example (Figure 2.1) shows that the input text represents the desired relation between the manufacturing features using the expression "should". This would require a different kind of processing as compared to the statement that uses "preferred to" to represent a rule about preference.

## 2.4 Overview of Formal Rule Extraction Framework

Figure 2.2 shows the overview of the formal manufacturing rule extraction framework. As mentioned in the previous section, the framework accepts the input of a manufacturing text with rule-like information and generates the equivalent formal rule written in SADL [11]. The entire process is divided into the following two stages: manufacturing knowledge construction, shown as the dotted line region in Figure 2.2, and rule extraction stage, shown as the solid line region in Figure 2.2. Manufacturing knowledge construction stage is the preliminary stage where the basis of the formal manufacturing rule extraction is formed, and rule extraction stage is the main stage where each sentence of the input text is translated into the equivalent formal manufacturing rule.

**Manufacturing knowledge construction stage** Manufacturing knowledge construction stage is the one-time process performed before rule extraction to form the basis of formal rule extraction. First, from the whole input text, important manufacturing-related terms are mapped into the smaller set of manufacturing concepts consisting a controlled vocabulary. Then each concept in the controlled vocabulary is converted to either formal class or property consisting the manufacturing domain ontology. On the other hand, non-noun terms, such as verb and adjective, and implicit manufacturing relations are captured as the semantic relations between the concepts in the controlled vocabulary and then converted to formal properties to define the relations between the formal classes. As the basis of rule extraction, the controlled vocabulary and manufacturing domain ontology play important roles. Specifically,

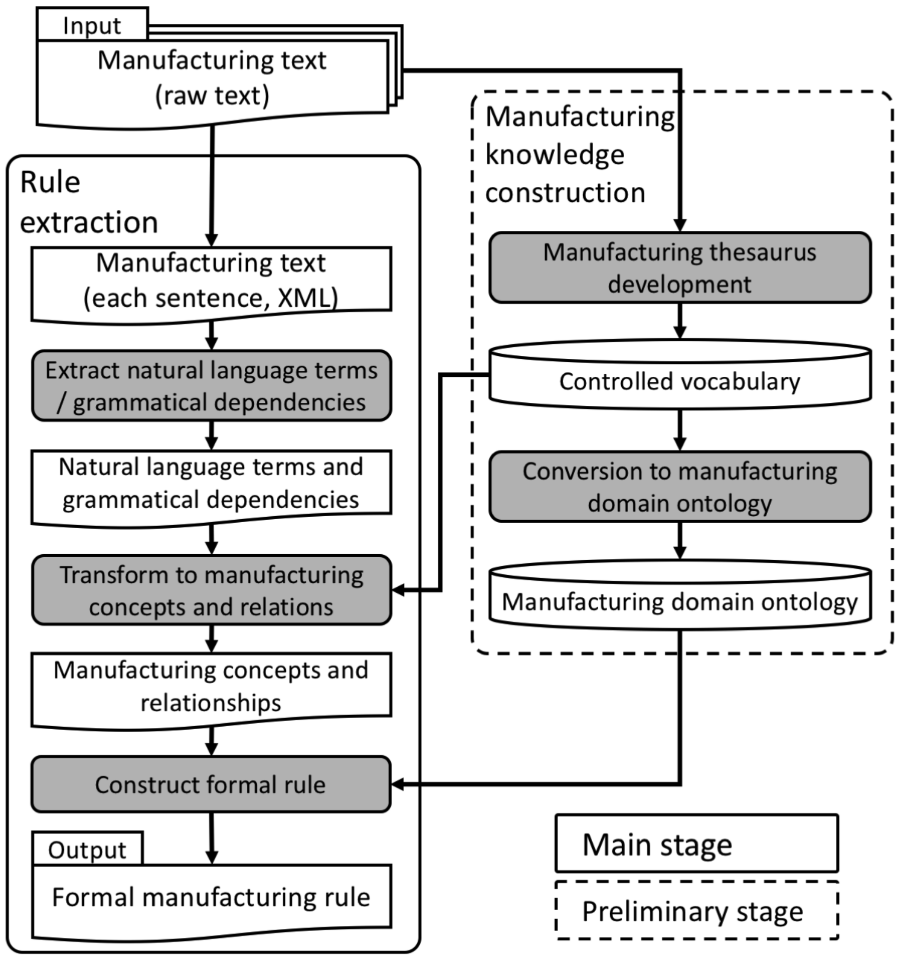



Figure 2.2: The overview of formal manufacturing rule extraction framework. The solid arrows indicate the sequence of execution. It should be noted that dotted line region represents one-time preliminary stage, which is manufacturing knowledge construction stage, and solid line region represents main stage, rule extraction stage, which translates each sentence of the input text into the equivalent formal manufacturing rule.

the controlled vocabulary provides the term-concept-class/property mapping when transforming natural language terms/grammatical dependencies to manufacturing domain. The manufacturing domain ontology ensures that the output formal rule conforms to the manufacturing domain.

**Rule extraction stage** Rule extraction stage is the main stage that each sentence of the input manufacturing text is translated into the equivalent formal manufacturing rule conforming to the manufacturing domain ontology. Before rule extraction stage, the input text is manually preprocessed into each sentence in XML [76] that also contains the section titles that each sentence belongs to. The preprocessing enables the framework to consider not only the content of the input text but also the section titles in rule extraction. After the preprocessing, the framework extracts natural language terms and grammatical dependencies utilizing standard NLP techniques. Then the framework transforms the natural language terms/grammatical dependencies to manufacturing concepts/relations based on the mapping provided by the controlled vocabulary. Lastly, from the manufacturing concepts/relations, the framework constructs the coherent formal manufacturing rule, i.e., the formal rule contains all the necessary manufacturing concepts/relations conforming to the manufacturing domain ontology.

## 2.5 Manufacturing Knowledge Construction Stage

The objective of this stage is to construct a controlled vocabulary and manufacturing domain ontology from input texts to form the basis of formal rule extraction. Figure 2.3 shows the manufacturing knowledge construction performed for the motivating example and another similar manufacturing text. Specifically, the important nouns such as “entry surface”, “entrance surface”, and “hole” (the terms in square brackets in Figure 2.3) are mapped to the concepts *entranceSurface* and *hole* to develop the controlled vocabulary, which is shown as the dotted line region in Figure 2.3. Then, the concepts are converted to the formal classes *EntranceSurface* and *Hole* consisting the manufacturing domain ontology, which is shown as the solid line region in Figure 2.3. The hierarchies between the terms, such as that betwen “surface” and “entrance surface”, are mapped to the semantic relation *narrower* in the controlled vocabulary, and the relation is converted to the formal property *subclass* in the manufacturing domain ontology. The important non-noun terms, phrasal verbs such as “be perpendicular to”, and the relations represented with prepositions between nouns such as “entrance surface of a hole” (the terms in parentheses in Figure 2.3) are also mapped to the semantic relations *related* in the controlled vocabulary. Then the relations are converted to the formal properties *isPerpendicularTo* and *hasEntranceSurface* in the manufacturing domain ontology. It should be noted that the methodology suggested by Ameri et al. [61] is adopted in creating the controlled vocabulary and manufacturing domain ontology. The brief introduction of each step is as follows.

**Manufacturing thesaurus development** In this step, manufacturing-related nouns and compound nouns are manually collected from the input text and regarded as the possible labels of manufacturing concepts. For example, Figure 2.3 shows that terms including “surface”, “entrance surface”, and “hole” are extracted and become the labels of the concepts *surfacce*, *entranceSurface*,

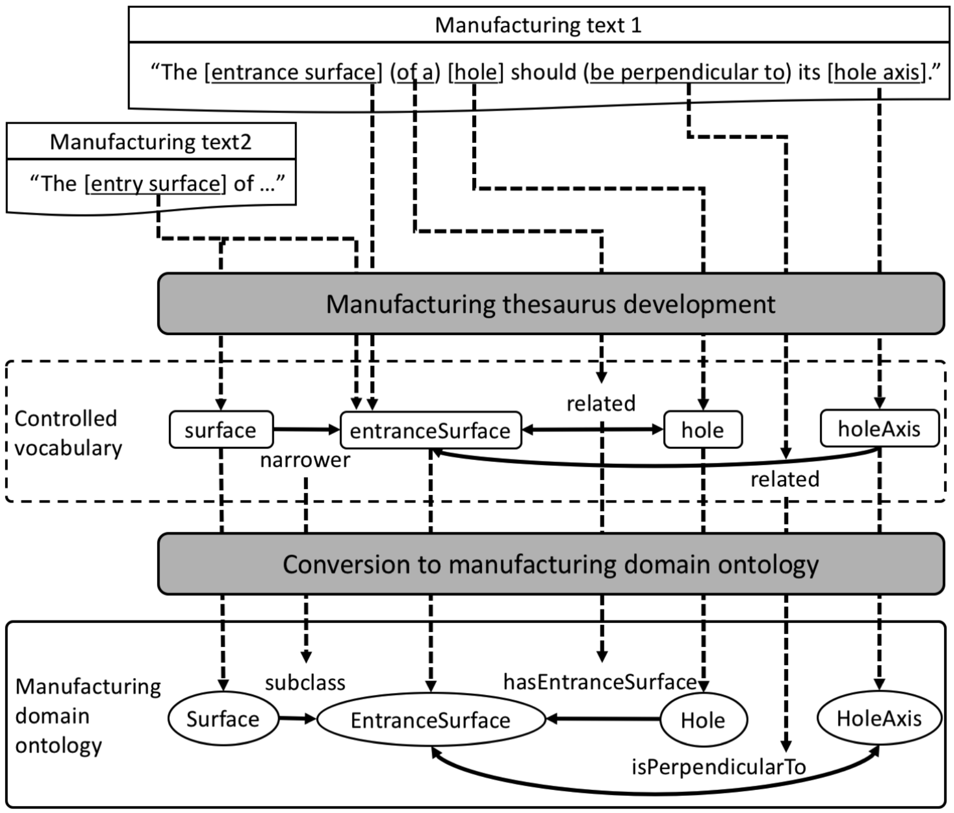



Figure 2.3: Manufacturing knowledge construction stage performed for the input text of the motivating example (shown at the top of Figure 2.1) and another similar text. The dotted line region contains the controlled vocabulary and solid line region contains the manufacturing domain ontology. The dotted arrows indicate the mapping from important manufacturing terms (e.g. "entrance surface" and "hole") to manufacturing concepts (e.g. *entranceSurface* and *hole*), and further mapping from the concepts into formal classes (e.g. *EntranceSurface* and *Hole*) of the manufacturing domain ontology, and the solid arrows indicate the semantic relation (e.g. *narrower* and *related*) among the concepts in the controlled vocabulary and the formal properties (e.g. *subclass* and *hasEntranceSurface*) defined between the formal classes.

and *hole* consisting the controlled vocabulary. When multiple terms refer to the same concept, a concept in the controlled vocabulary can have multiple terms as its labels. For example, Figure 2.3 shows that both “entry surface” and “entrance surface” refer to the same concept *entranceSurface*, therefore, one is used as the *prefLabel* (i.e., preferred label) and the other one is used as the *altLabel* (i.e., alternative label) of the same concept. In addition, the semantic relations between the concepts are defined as the following three types: *narrower*, *broader*, and *related*. For example, Figure 2.3 shows that the concept *entranceSurface* is a *narrower* concept of another concept *surface* since “entrance surface” is a specific type of “surface”. On the other hand, *entranceSurface* is a *related* concept of the concept *hole* since the two nouns “entrance surface” and “hole” are possibly related through “be perpendicular to”. Once the controlled vocabulary is created, it is reviewed by domain experts for semantic consistency.

**Conversion to manufacturing domain ontology** In this step, the controlled vocabulary is converted to the manufacturing domain ontology, which is the foundation of the rule layer that is the primary focus of formal manufacturing rule extraction. Specifically, the concepts and the semantic relations in the controlled vocabulary are extended to the formal classes or formal properties of the manufacturing domain ontology. For example, Figure 2.3 shows that the formal classes *Surface*, *EntranceSurface*, and *Hole* are created by extending the concepts *surface*, *entranceSurface*, and *hole*. Similarly, the formal properties *subclass* and *hasEntranceSurface* are created by extending the semantic relations *narrower* and *related* between the concepts of the controlled vocabulary respectively.

It should be noted that the controlled vocabulary and manufacturing domain ontology are manually constructed to represent all the manufacturing concepts and relations in the input manufacturing text. In other words, all the terms related to manufacturing are chosen for the controlled vocabulary and converted to the formal classes or properties in the manufacturing domain ontology. For a larger scale example, additional processes including systematic concept screening and concept validation are required to facilitate the creation of a controlled vocabulary and manufacturing domain ontology. For the further details of these steps, the readers are referred to the literature [61]. The following subsections describe how the controlled vocabulary and manufacturing domain ontology provide the basis of the rule extraction stage. First, once constructed, the controlled vocabulary provides the mapping from natural language terms/grammatical dependencies to manufacturing concepts/relations. The manufacturing domain ontology provides the formal representation to create a coherent formal manufacturing rule which is a formal rule conforming to the domain and containing all the necessary manufacturing concepts and relations.

### 2.5.1 Controlled Vocabulary as the Bridge between Text and the Manufacturing Domain Ontology

To create a coherent formal manufacturing rule from a text, it is critical to ensure that the breadth of natural language terms is addressed with a smaller set of formal classes or properties. For example, some of the manufacturing text can be written using the phrase "entrance surface" while others may use the different phrase "entry surface." Without properly mapping the terms, they are regarded as pointing to the different formal classes or properties that should have been bound to the same formal class or property in the domain ontology. However, a formal class or property cannot have multiple natural language terms as its attributes due to the limitation of the standard of Web Ontology Language (OWL) [10]. Therefore, the mapping between them should be defined outside of the manufacturing domain ontology and application code. In short, the "bridge" between the natural language terms and the formal classes/properties is required. In this dissertation, the controlled vocabulary based on Simple Knowledge Organization System (SKOS) [77] is used to perform the above-mentioned role. In SKOS-based controlled vocabulary, natural language terms are regarded as the labels for the concepts while semantic relations are defined between the concepts. In other words, the information is organized on a conceptual level, not in lexical level. Such concept-based nature enables a concept of the controlled vocabulary to own natural language term(s) as its label(s) and facilitates the concept to be converted to a formal class or property consisting the manufacturing domain ontology. In short, once the controlled vocabulary and manufacturing domain ontology are created from the natural language terms, the term-concept-class/property mappings shown in Figure 2.3 (e.g. "hole"-*hole*-*Hole*) are automatically defined. It should be noted that multiple terms can be mapped to the same formal class or property with the approach. For example, Figure 2.3 shows that the natural language terms "entrance surface" and "entry surface" are mapped into the same concept *entranceSurface* and further mapped to the formal class *EntranceSurface* to specify that the terms represent the same manufacturing concept. In addition to as the solution to the mapping issue, the explicit mapping enabled by the controlled vocabulary is favorable since the controlled vocabulary is reusable and can be easily extended without modifying or recompiling the entire framework. It should be noted the mapping is defined between the nouns/compound nouns and formal classes/properties. The non-noun terms including verbs and adjectives are directly matched from the natural language terms to the formal properties.

### 2.5.2 Domain Ontology as the Building Block of a Formal Manufacturing Rule

In this dissertation, the manufacturing domain ontology is utilized to provide the formal classes and properties to represent the formal statements so that the framework can construct a coherent formal manufacturing rule, i.e., the formal rule that contains all the necessary concepts and their relations conforming to manufacturing domain. In other word, the manufacturing domain ontology is used as the building block to represent a formal manufacturing rule. For example, Figure 2.1 shows that formal classes and formal properties are used to define the instances and their relations in the formal statements consisting the desired formal manufacturing rule (e.g. "*entrance_surface* is a *EntranceSurface*" and "*entrance_surface isPerpendicularTo hole_axis*"). Figure 2.4 shows a part of the manufacturing domain ontology used in this dissertation. The manufacturing domain ontology includes the formal classes (shown as ovals) corresponding to the manufacturing features such as *Hole*, *EntranceSurface*,

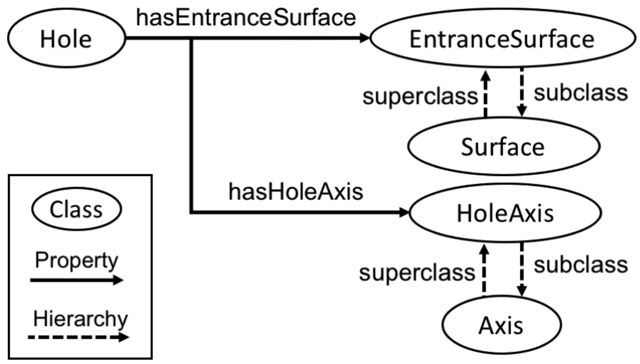



Figure 2.4: A part of the manufacturing domain ontology used in this chapter corresponding to the motivating example. The ovals indicate the formal classes, and the solid lines indicate the formal properties, and the dotted lines indicate the hierarchy between the formal classes.

and *HoleAxis* along with their hierarchy (shown as the dotted lines), and formal properties (shown as the solid lines) such as *hasEntranceSurface* and *hasHoleAxis*. In this dissertation, Manufacturing Service Description Language (MSDL) [2] is used as the upper ontology serving as the unifying model so that the lower-level classes and properties created in this stage can be connected each other.

## 2.6 Rule Extraction Stage

Rule extraction stage is the main stage that each sentence of manufacturing texts is translated into the coherent formal manufacturing rule. The stage includes the three steps shown in Figure 2.2. Specifically, the framework extracts natural language terms/grammatical dependencies using standard NLP techniques, transforms the natural language terms/dependencies to manufacturing concepts/relations, and constructs a formal manufacturing rule from the manufacturing concepts/relations. The details are explained in the following subsections.

### 2.6.1 Extract Natural Language Terms / Grammatical Dependencies

Computer manipulations involving "understanding" natural language are referred as Natural Language Processing (NLP) [78]. In this step, NLP is the core mechanism to interpret a manufacturing text. Specifically, standard NLP techniques including POS tagging, co-reference resolution, and dependency parsing are utilized to break each sentence into natural language terms and derive the grammatical dependencies between the terms. The grammatical dependencies provide the structured information of the input sentence so that a formal manufacturing rule can be constructed in the later steps. Figure 2.5 shows natural language processing step performed for the motivating example (Figure 2.1). Specifically, the framework breaks the input sentence into separate terms (e.g. "the", "entrance", "surface" . . . ) and derives the grammatical dependencies between the terms (e.g. "nsubj" for nominal subject and "pobj" for prepositional object). The full list of grammatical dependencies is shown in Appendix B of this dissertation. Another task done in this step is co-reference resolution that derives the term referred by another term such as a pronoun. In the motivating example, the pronoun "its" refers to "hole", and this co-reference is denoted as the grammatical dependency "coref" in Figure 2.5. In this dissertation, two NLP libraries are used. NLP4J [79] is used to derive the natural language terms/grammatical dependencies, and OpenNLP [80] is used for co-reference resolution.

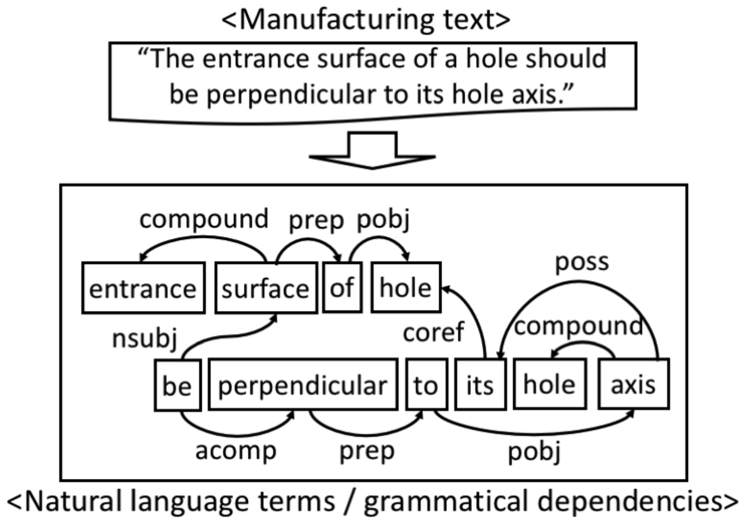



Figure 2.5: Natural language terms and grammatical dependencies derived from the motivating example (Figure 2.1) are shown. The nodes of the graph indicate the separated natural language terms and the connections between the nodes indicate the grammatical dependencies between the terms. In addition, the co-reference between the terms "its" and "hole" is also derived and denoted as the dependency "coref".

### 2.6.2 Transform to Manufacturing Concepts and Relations

The natural language terms/grammatical dependencies derived in the previous step do not conform to manufacturing domain. In this step, the framework transforms the natural language terms/grammatical dependencies to manufacturing domain by aligning them into formal classes and properties in the manufacturing domain ontology. Figure 2.6 shows the transformation step performed for the motivating example. The nouns and compound nouns (multiple terms that are interpreted as one term or concept such as "entrance surface" and "hole axis", shown as shaded regions in Figure 2.6) are identified based on the controlled vocabulary and transformed to the manufacturing concepts corresponding to the formal classes or properties. On the other hand, the non-noun terms such as adjective and phrasal verbs (a phrase consisting of a verb and other elements giving a meaning different from the verb such as "is perpendicular to", also shown as shaded region in Figure 2.6) are identified from grammatical dependencies and directly transformed to the corresponding formal properties. For example, Figure 2.6 shows that the phrase "is perpendicular to" is directly matched to the formal property *isPerpendicularTo* in the manufacturing domain ontology. It should be noted that the transform of non-noun terms and phrasal verbs are not done by the controlled vocabulary since they are represented as one of the three semantic relations (i.e., *narrower*, *broader*, and *related*) in the controlled vocabulary, and the mapping between semantic relations and formal properties are not defined in the controlled vocabulary. Similarly, implicit relations such as *hasEntranceSurface* and *hasHoleAxis* are directly derived from the grammatical dependencies.

### 2.6.3 Construct Formal Rules

In this step, from the manufacturing concepts and relations, the framework constructs a formal manufacturing rule conforming to the manufacturing domain ontology. Specifically, the formal classes/properties corresponding to the manufacturing concepts/relations become one of the subject, predicate, or object of a formal statement (triple). Figure 2.7 shows the process of constructing the formal manufacturing rule. A formal manufacturing rule has two parts: Condition part that represents the impediment for manufacturing (shown as the solid line region at the top-right in Figure 2.7), and triggered action part that contains feedback for fixing the manufacturability issue (shown as the dotted line region at the bottom-right in Figure 2.7).

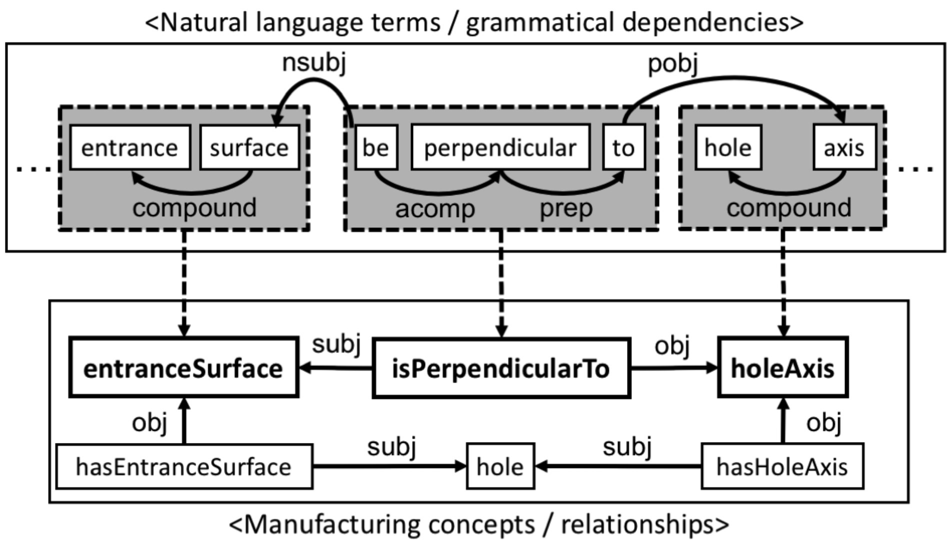



Figure 2.6: Natural language terms/grammatical dependencies of the motivating example are transformed to manufacturing concepts/relations. Compound nouns (nouns that consist of multiple terms such as "entrance surface" and "hole axis" shown as the regions) and phrasal verbs (idiomatic phrases that consist of a verb and another element such as "be perpendicular to" shown as the shaded region) are identified and the node consisting them are merged to form manufacturing concepts and relations corresponding to the formal classes and properties in the manufacturing domain ontology.

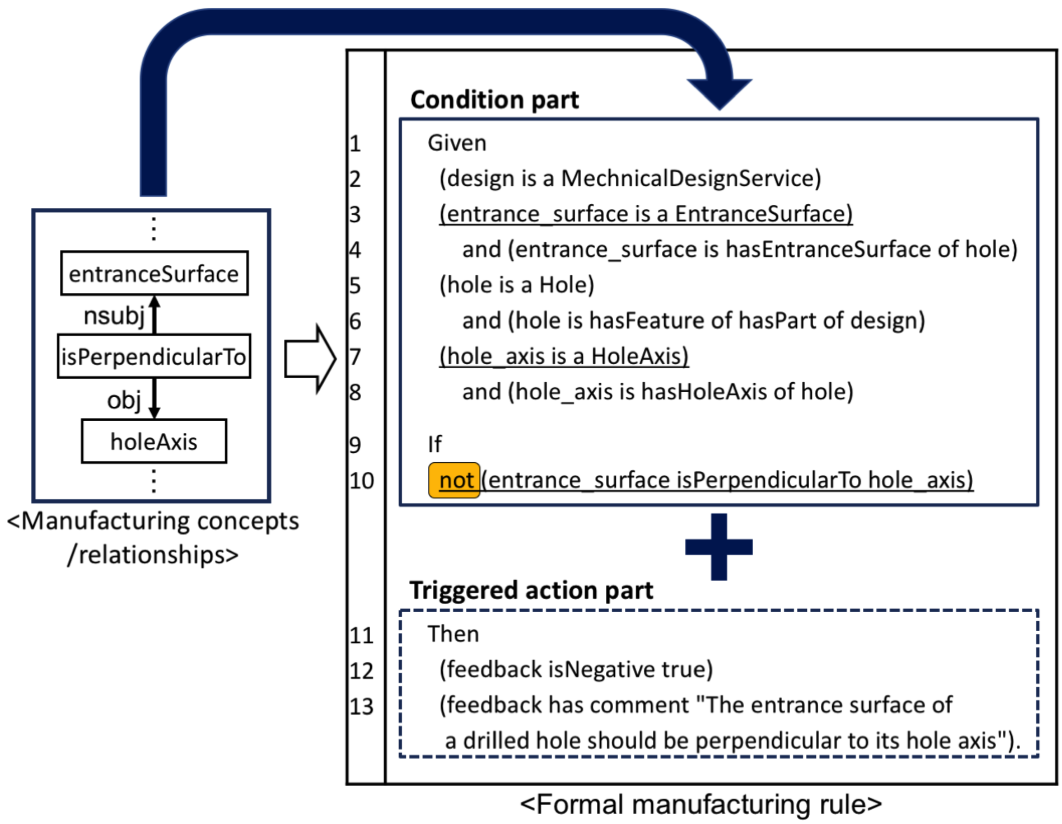



Figure 2.7: The formal manufacturing rule constructed from the manufacturing concepts/relations derived from the motivating example (Figure 2.1) is shown. The rule has two parts: the condition part that represents the impediment for manufacturing, and the triggered action part that contains the feedback for addressing the manufacturability issue.

**Condition part of the rule** The condition part of the rule, which is shown after the markers *Given* and *If*, is created from the manufacturing concepts/relations to represent the impediments for manufacturing. Specifically, the types of instance are defined using the formal classes such as *EntranceSurface* and *HoleAxis*, and their semantic relations are represented using the formal properties such as *isPerpendicularTo*. To make a coherent formal manufacturing rule, the framework considers two things: the formal manufacturing rule model and the type of the manufacturing rule. First, the framework utilizes the formal manufacturing rule model that a formal manufacturing rule should conform to. Figure 2.8 shows the entity-relationship diagram (ERD) of the model. Based on the model, the framework examines if a lower-level entity is declared without higher-level entity, such as the declaration of a *Feature* without corresponding *Part*. In that case, the missing entity is declared to make the rule coherent. In the example, *Part* is declared as "*hasPart of design*" (line 6 of Figure 2.7) as it is higher-level entity of *Feature* including *EntranceSurface* and *Hole*. Also, the highest entity *Design* is declared to encapsulate all the other entities, as shown in line 2 of Figure 2.7. By using the rule model, the framework can capture missing or implicit manufacturing concepts/relations so that the constructed rule contains all the necessary concepts. Once the formal classes and properties related to the formal rule are identified, the framework chooses different representations of condition part based on the type of the manufacturing rule. In this dissertation, two types of representations are addressed. The first type is the manufacturing rule that explicitly specifies the constraint that a design should satisfy. This type of manufacturing rule is represented by using the expression including "if . . . then" and "should" as the motivating example. In this case, the framework simply includes all the formal classes/properties in the condition part and puts *not* in front of the main statement, which is the formal statement related to the main verb. By putting *not* in front of the main statement, the rule can identify the impediment for manufacturing. Figure 2.7 shows one of the example. As the rule has *not* in front of the main statement "*entrance_surface isPerpendicularTo hole_axis*", it can identify the impediment for manufacturing. On the other hand, the second type is the manufacturing rule that explicitly specifies the preference of a design over another design. For example, the manufacturing text that contains the expressions such as "A prefer to B" and "A is more economical than B" is regarded as the second type. In this case, the framework creates the formal rule that can identify the adoption of the less desired design. For example, the rule extracted from the sentence "round part is more economical than square part" only contains the formal statement "*part isSquare true*" in its condition part even though "round part" is also mentioned in the text. In that way, the rule can identify if the less desired design "square part" is chosen.

**Triggered action part** On the other hand, the triggered action part of a formal manufacturing rule, which is shown after the marker *Then*, contains the feedback and comment when the condition part is triggered, i.e., when the given design has the impediments for manufacturing. Specifically, when the given design satisfies the condition part of the rule in Figure 2.7, the rule provides the negative feedback and the comments "the entrance surface of a

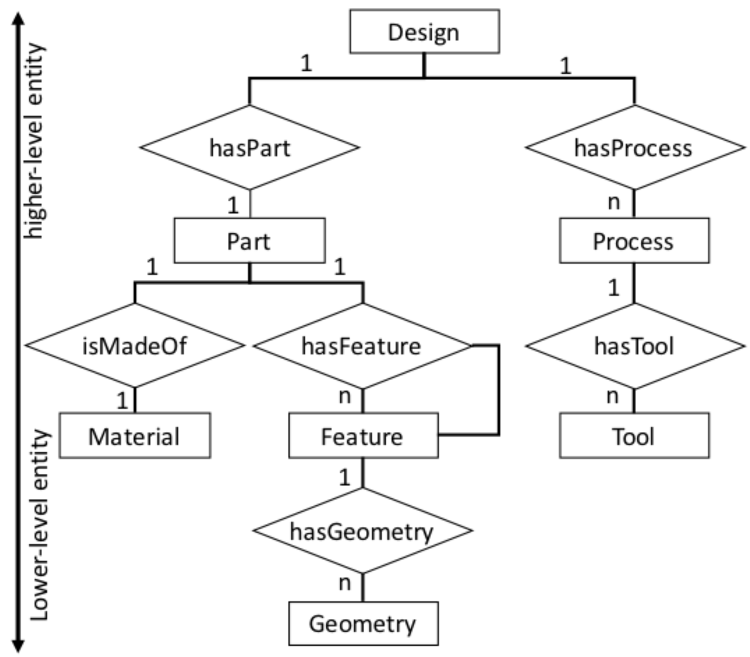



Figure 2.8: The entity-relationship diagram (ERD) of the formal manufacturing rule model is shown. The rule extraction framework utilizes the model in two ways. First, the framework create the highest concept *Design* to encapsulate all the entities in a formal manufacturing rule. Second, when a lower-level entity is declared without its corresponding higher entity (e.g. a *Feature* is declared without *Part*), the framework also declares the higher entity in the formal manufacturing rule so that implicit relations between the concepts can be captured.

drilled hole should be perpendicular to its hole axis" so that the manufacturability issue can be addressed accordingly.

## 2.7 Implementation and Experiment

### 2.7.1 Preparation of the Dataset

The dataset was created from the design for manufacturing handbook written by Bralla [12]. First, the test dataset, which consists of 36 sentences in milling, metal stamping, and die-casting sections, is created for developing the framework. Once the framework is developed, the validation dataset is created to verify the extensibility of the framework. The validation dataset contains 133 sentences from the 30 sections in forming, machining, casting, molding, assembling, and finishing chapters of the book. In creating the dataset, sentences that appeared to capture manufacturing rules were manually selected. Then some of the sentences were modified or discarded to ensure that the dataset conforms to the manufacturing domain ontology and NLP results are valid. For more details about the dataset creation and the list of sentences in the dataset, the readers are referred to Appendix A of this dissertation.

As mentioned, the input text is separated into each sentence and preprocessed into XML [76]. Figure 2.9 shows a part of the input XML file corresponding to the motivating example (Figure 2.1). The XML input also contains the section titles that each sentence belongs to. When a sentence does not explicitly specify the context, such as the type of manufacturing process or material, the framework searches the section titles to derive the context.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!-- Input (motivating example) -->

<book name="DESIGN FOR MANUFACTURABILITY HANDBOOK" author="James G. Bralla">
    <!-- # of sentence in this part: 16 -->
    <part id="4" name="MACHINED COMPONENTS">
        <chapter id="6" name="PARTS PRODUCED ON MILLING MACHINES">

            <sentence isRule="true">
                The entrance surface of a hole should be perpendicular to its hole axis.
            </sentence>
        </chapter>
    </part>
</book>
```

Figure 2.9: The XML [76] input corresponding to the motivating example (Figure 2.1) is shown. It also contains the section titles that the sentence belongs to so that the framework can derive the context not explicitly mentioned in the sentence.

### 2.7.2 Construction of Controlled Vocabulary and Manufacturing Domain Ontology

From the dataset, manufacturing knowledge construction is performed to provide the controlled vocabulary and manufacturing domain ontology for rule extraction. As both SKOS [77] and OWL [10] are W3C recommendation built upon Resource Description Framework (RDF) [81]/Resource Description Framework Schema (RDFS) [82], the controlled vocabulary and manufacturing domain ontology can be encoded using the same environment. In this dissertation, Semantic Application Design Language (SADL) [11] is used to encode the controlled vocabulary and manufacturing domain ontology as shown in Figure 2.10. Since it has English-like syntax, even domain experts not familiar with semantic technology can update the controlled vocabulary and manufacturing domain ontology when the application domain needs to be extended. In addition, as SADL [11] supports automated conversion to OWL [10] files, the controlled vocabulary and manufacturing domain ontology, written in SADL, are automatically converted to OWL [10] files that the framework can import.

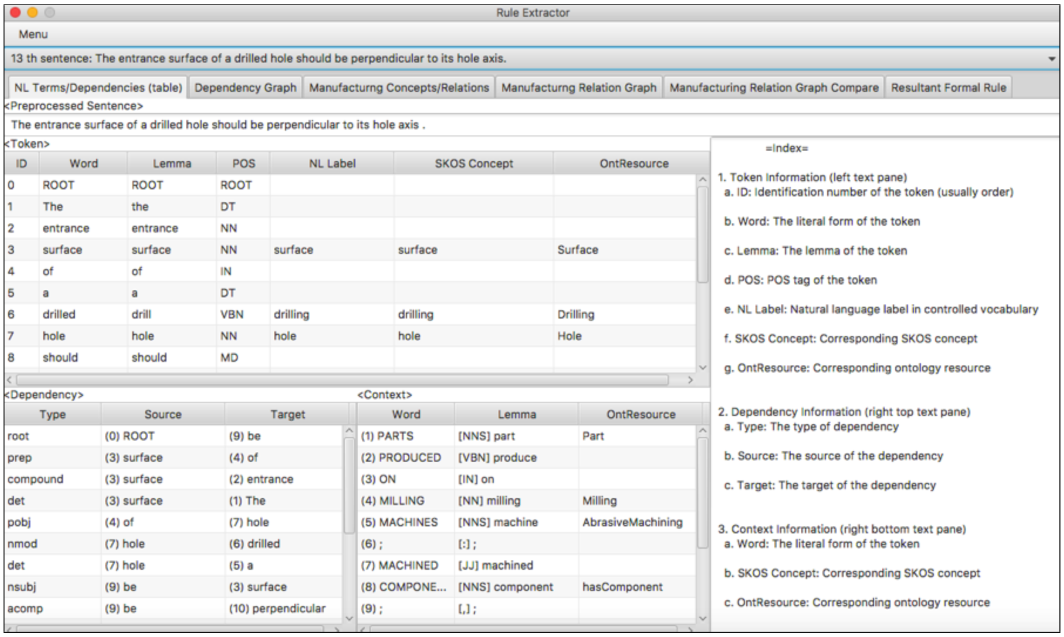
### 2.7.3 Implementation

To demonstrate the feasibility of the formal manufacturing rule extraction, a Java application is developed to implement the rule extraction framework. Figure 2.11 shows the screenshots of the Java application. The application utilizes various tools to extract formal manufacturing rules. First, the application utilizes NLP4J [79] and OpenNLP [80] to perform NLP. Second, the application utilizes SADL [11] to encode the controlled vocabulary, manufacturing domain ontology, and formal manufacturing rules. SKOS [77] is adopted as the representation for the controlled vocabularies. Lastly, the management of the controlled vocabulary/manufacturing domain ontology and reasoning process were performed by using Apache Jena Ontology API [83]. As mentioned, the controlled vocabulary, manufacturing domain ontology, and formal manufacturing rules written in SADL [11] are automatically translated to OWL [10] and Jena rules so that they can be imported using Jena Ontology API [83]. The user interface of the application provides the intermediate processes of rule extraction including natural language terms/grammatical dependencies, manufacturing concepts/relations, and output formal manufacturing rule. Therefore, the user can understand how a formal manufacturing rule is developed and find out if there is an issue. Especially, the natural language terms/grammatical dependencies and manufacturing concepts/relations are visualized in graph structure to facilitate the debug process.

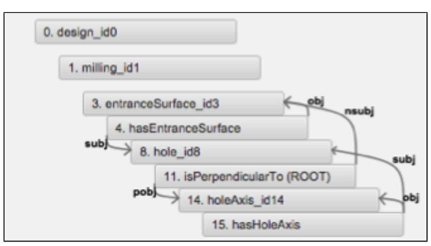
```
// Controlled vocabulary (motivating example)
manufacturingFeature is a Concept.
manufacturingFeature prefLabel "manufacturing feature".
manufacturingFeature altLabel "feature".
    surface is a Concept.
    surface broader manufacturingFeature.
    surface prefLabel "surface".
    surface related axis.
            entranceSurface is a Concept.
            entranceSurface broader surface.
            entranceSurface prefLabel "entrance surface".
            entranceSurface altLabel "entry surface".

    hole is a Concept.
    hole broader manufacturingFeature.
    hole prefLabel "hole".
    hole related entranceSurface.
    hole related holeAxis.

    axis is a Concept.
    axis prefLabel "axis".
        holeAxis is a Concept.
        holeAxis broader axis.
        holeAxis prefLabel "hole axis".
```

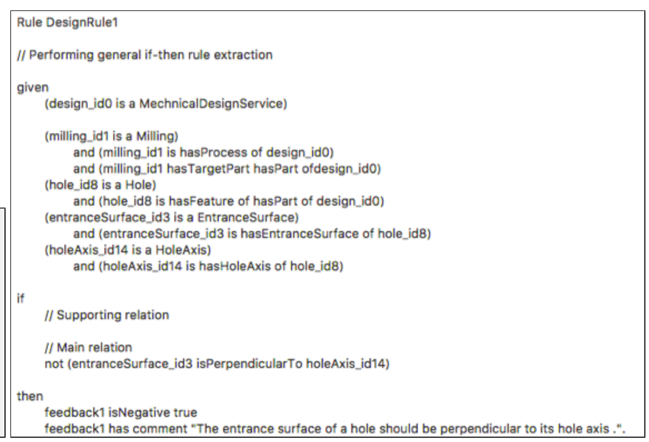
```
// Domain ontology (motivating example)
Surface is a type of MSDLGeometricClass.
    EntranceSurface is a type of Surface.

ManufacturingFeature is a type of MSDLCoreClass.
    Hole is a type of ManufacturingFeature.

Axis is a type of MSDLGeometricClass.
    HoleAxis is a type of Axis.

    hasAxis is a type of hasGeometry, describes Hole with values of type Axis.
    hasHoleAxis is a type of hasAxis, describes Hole with a single value of type HoleAxis.

isPerpendicularTo describes Surface with a single value of type Axis.
```

Figure 2.10: Parts of controlled vocabulary (top) and manufacturing domain ontology (bottom) used in this dissertation are shown. The parts correspond to the manufacturing concepts and relations in the motivating example (Figure 2.1). Both of them are written in Semantic Application Design Language (SADL) [11], which provide English-like language to encode semantic model




(a)




(b) (c)

Figure 2.11: The Java application implementing the proposed framework is shown. Specifically, the screenshots show the following intermediate processes: (a) NLP results shown with the overall user interface, (b) manufacturing concepts/relations visualized with graph structure, and (c) The output formal manufacturing rule.

### 2.7.4 Result and Discussion

The Java application was applied to the validation dataset, and the experimental result was verified by domain experts. The result demonstrates the feasibility of the framework by extracting 76 correct rules (about 57%) from 133 sentences. For the more detailed statistics of the rule extraction results and the full list of the extracted formal rules, the readers are referred to Appendix C and Appendix D of this dissertation respectively.

During the experiment, the extensibility of the framework is also verified by initially developing the framework using the test dataset and later extending to the validation dataset. The extension process only requires updating the controlled vocabulary and manufacturing domain ontology to address the additional application domains.

While the framework could extract the desired formal manufacturing rules from more than a half of the sentences in the validation dataset, several issues should be addressed to acquire correct rules from the rest of the cases. One of the important issues is the ambiguity caused by domain-specific terms, called as domain-specific ambiguity, which lead to 17 cases (approx. 13% of the dataset). In the rest of this section, the successful case of rule extraction and the unsuccessful case due to the above-mentioned issue are presented.

**Manufacturing rule is successfully extracted** In this case, the formal manufacturing rule is extracted and the semantics of the rule is verified. As mentioned, even though different rules with the equivalent semantics may ex-

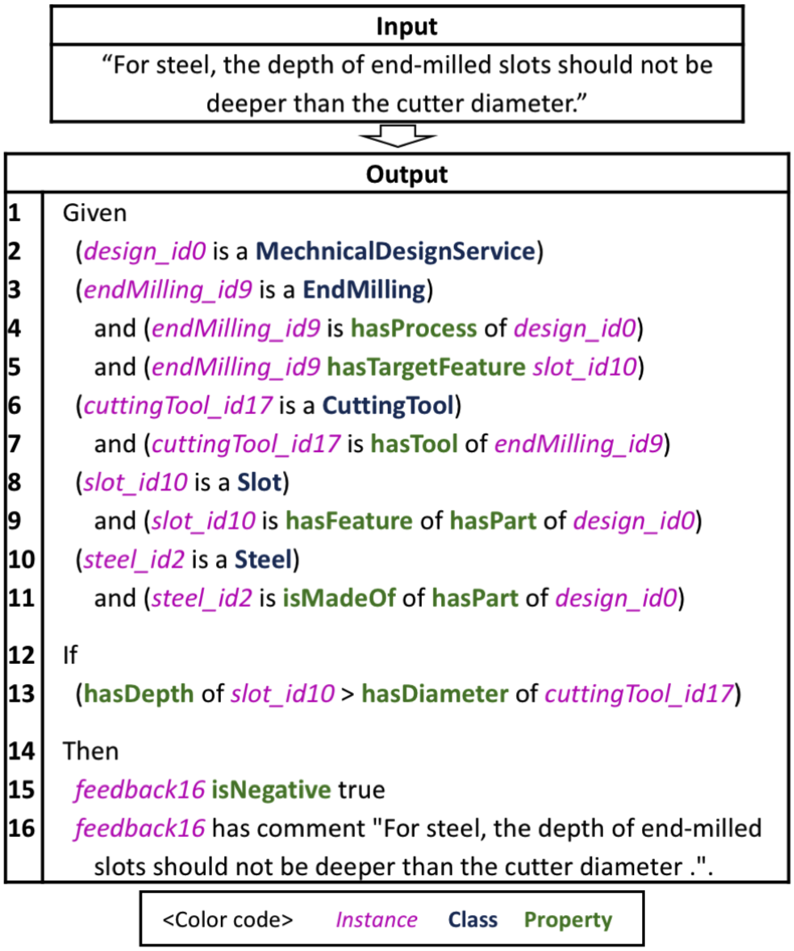



Figure 2.12: The successful case of formal manufacturing rule extraction is shown.

ist, the uniqueness of the representation is not considered in this dissertation. Figure 2.12 shows one of the successful cases. It should be noted that the framework substitutes the phrase “be deeper than” with the comparison operator “>” since the phrase is used to compare numeric values. Similar phrases such as “exceed” are also substituted with the corresponding comparison operator.

**Manufacturing rule is extracted, but the rule is not valid due to ambiguity of the sentence** In this case, the extracted rule is not valid, or the semantics of the rule is not correct due to the ambiguity of the sentence. From the 57 unsuccessful cases, 17 of them belong to this category (approx. 13% of the dataset). Figure 2.13 shows one of the cases. The input sentence has two types of ambiguities: coordination ambiguity and anaphoric ambiguity. First, coordination ambiguity occurs when the phrases connected with conjunctions are modified, like the phrase "slot widths, radii, and chamfers" in the example. Specifically, while the phrase implies that the term "slot" modifies all the following nouns "widths", "radii", and "chamfers", the correct interpretation was not derived by standard NLP techniques. That is the reason the ownership between *slot_id2* (line 8 of Figure 2.13) and *chamfer_id8* (line 10 of Figure 2.13) is missing in the output rule. Second, anaphoric ambiguity occurs when there are multiple choices of antecedents for an anaphor, which is a referring term such as a pronoun, like the pronoun "those" in the example. Specifically, while the pronoun refers to the terms "widths", "radii", and "chamfers", standard NLP techniques could not derive the correct referred terms. That is the reason the output rule has *those_unclassified*, which is the unclassified instance shown in line 15-17 of Figure 2.13. As standard NLP techniques can address some of the ambiguities in a general text, these cases imply that standard NLP techniques failed to address these ambiguities due to the lack of considering the domain context. For example, the first ambiguity can be resolved by considering the domain context that "widths", "radii", and "chamfers" can be the attributes of a "slot". Similarly, the second ambiguity can be resolved by considering the domain context that the phrase "those of standard cutters" implies that the pronoun "those" refers to "widths, radii, and chamfers" that can be the attributes of a "cutter". Resolving such ambiguities, called domain-specific ambiguity since it is related to the domain context, will be addressed in Chapter 3 of this dissertation to extend the rule extraction framework.

**Manufacturing rule is not extracted properly** In this case, rule extraction was not successful due to the other issues including complex sentence structure and the inclusion of unnecessary information. For example, semantically invalid rule was extracted from the sentence "It is better not to have sharp corners on a formed part" as standard NLP techniques could not derive the subject of the sentence. Specifically, the subject of the sentence is dummy pronoun "it", and addressing such dummy pronoun requires re-writing the sentence that is not feasible with standard NLP techniques. In this dissertation, such cases are addressed in Chapter 4 by providing the relevant feedback so that user can address the issue to acquire correct rules. It should be noted that improving NLP techniques to address the issues is out of the scope of this dissertation.

## 2.8 Summary

In this chapter, the extraction of formal manufacturing rule from unstructured natural language text is presented. Specifically, the semantic-based framework is developed to extract formal manufacturing rules conforming to the manufacturing domain ontology. In addition, the controlled vocabulary is used to

**Input**

"Slot widths, radii, and chamfers should conform to those of standard cutters."

**Output**

```
1  Given
2    (design_id0 is a MechnicalDesignService)
3    (milling_id1 is a Milling)
4      and (milling_id1 is hasProcess of design_id0)
5      and (hasPart of design_id0 is hasTargetPart of milling_id1)
6    (cuttingTool_id18 is a CuttingTool)
7      and (cuttingTool_id18 is hasTool of milling_id1)
8    (slot_id2 is a Slot)
9      and (slot_id2 is hasFeature of hasPart of design_id0)
10   (chamfer_id8 is a Chamfer)
11     and (chamfer_id8 is hasFeature of hasPart of design_id0)

12 If
13   (cuttingTool_id18 isStandard true)
14   (
15     not (hasWidth of slot_id2 == those_unclassified) or
16     not (hasRadius of null == those_unclassified) or
17     not (chamfer_id8 == those_unclassified)
18 )

19 Then
20   feedback2 isNegative true
21   feedback2 has comment "Slot widths, radii, and chamfers
       should conform to those of standard cutters .".
```

<Color code> *Instance* **Class** **Property**

Figure 2.13: The unsuccessful case of formal manufacturing rule extraction is shown. Due to the unresolved ambiguities, the extracted rule does not define the ownership between *slot_id2* (line 8) and *chamfer_id8* (line 10) and contains the unresolved pronoun *those_unclassified* (line 15-17), shown as shaded box.

map natural language text into manufacturing domain, and the formal manufacturing rule model is used to construct coherent formal manufacturing rule. The feasibility of the framework is also verified from the dataset created from a design for manufacturing handbook [12]. Specifically, precise formal manufacturing rules are extracted from more than a half of the sentences in the dataset, except for the cases when the input text contains ambiguity, or the cases related to the issues such as complex sentence structure and the inclusion of unnecessary information.

# CHAPTER 3: DOMAIN-SPECIFIC AMBIGUITY RESOLUTION OF MANUFACTURING TEXT

## 3.1 Motivation

The previous chapter demonstrated the formal manufacturing rule extraction framework and its feasibility. However, the experimental result implied the additional gaps to be addressed in a practical scenario. One of the gaps is the lack of domain-specific ambiguity resolution for manufacturing text. It is important since the extraction of the desired rule is not guaranteed when a manufacturing text has an ambiguity, i.e., can be interpreted in more than one way.

To address the above-mentioned gap, this chapter proposes the domain-specific ambiguity resolution method that complements standard NLP techniques with the domain context. Specifically, the method utilizes the manufacturing domain ontology as the mechanism to incorporate domain context in ambiguity resolution process. To demonstrate the effectiveness of the ambiguity resolution method, the previously developed rule extraction framework is extended to adopt the method and tested on the same dataset. The result validates the method by an increasing the correct rules to 70% of the 133 sentences, which is increased by approx. 13%.

The remainder of this chapter is organized as follows. First, Section 3.2 reviews the existing works relevant to ambiguity resolution. Then Section 3.3 defines the main problem of this chapter, and Section 3.4 presents the overview of the developed method. The detailed steps of the ambiguity resolution method are presented from Section 3.5 to Section 3.7. Lastly, the implementation of the extended rule extraction framework and experimental result are shown in Section 3.8, and Section 3.9 summarizes this chapter.

## 3.2 Relevant Work

Natural Language Processing (NLP) refers to any computer manipulations involving "understanding" natural language [78], and ambiguity resolution has been one of the challenging problems in NLP. Its objective is to resolve any ambiguities in a sentence that arises whenever an expression can be interpreted in more than one way [84]. In this section, the existing methods for resolving the two common types of ambiguities, coordination ambiguity and anaphoric ambiguity, are reviewed. Also, the challenges in ambiguity resolution of manufacturing text are presented based on the literature review.

### 3.2.1 Anaphoric Ambiguity Resolution

An anaphor (plural: anaphora) is a linguistic expression that refers to a preceding utterance in a text, such as the pronouns "it" and "they". Even though an anaphor is a useful grammatical tool that helps not to repeat the same utterance, it may lead to an ambiguity when there are multiple possible antecedents for the anaphor. This kind of ambiguity is referred as anaphoric ambiguity. The sentence "slot widths and radii should conform to those of cutters" is an example of the manufacturing text with anaphoric ambiguity. Specifically, it is ambiguous if the pronoun "those" refers to which of the antecedent nouns "slot", "widths", and "radii". In this section, various approaches for anaphoric ambiguity resolution are reviewed. For the broader overview of the anaphoric ambiguity resolution, the readers are referred to the literature [85]–[87].

Early approaches had relied on the heuristics based on lexical, syntactic, and semantic knowledge. The approaches include that of Lappin and Leass [88] and those of Mitkov [89], [90]. Lappin and Leass [88] developed the logic parser that utilizes discourse salience from syntactic structure, such as the distance between antecedents from anaphor and parallelism, to resolve anaphoric ambiguity. It successfully improved and employed previously designed algorithms including Hobbs' algorithm [91] and context mechanism by Alshawi [92] for anaphoric ambiguity resolution. Mitkov [89] also developed knowledge-based anaphoric ambiguity resolution model that integrates syntactic, semantic, discourse, domain, and heuristical knowledge for ambiguity resolution. On the other hand, the later approach of Mitkov [90], referred as knowledge-poor approach, is designed to minimize time-consuming parsing process. The approach utilizes a part-of-speech (POS) tagger, simple heuristics including gender/number agreement rule, the distances between references, and term preferences.

Other early approaches had employed computational theories of discourse such as centering algorithm [93]. The approaches include that of Brennan et al. [94], Strube [95], and Tetreault [96]. For example, Brennan et al. [94] used the centering approach to model the attentional structure in discourses, track discourse contexts, and bind unresolved pronouns. Strube [95] developed the S-list approach to address the computational overhead of the approach of Brennan et al. [94]. Strube suggested the concept of S-list that describes the attentional state of the hearer at any given point in processing a discourse so that ambiguity resolution can be performed by looking up the S-list. Tetreault [96] developed the algorithm based on Left-Right Centering algorithm [97] which also focuses on the incremental processing of utterances for lower computational overhead.

Recently, since annotated corpus became accessible and advanced Machine Learning (ML) methods are developed, supervised ML-based approaches and statistical approach have proven the better performance in anaphoric ambiguity resolution. For example, Dagan and Itai [98] collected the co-occurrence pattern of the terms and utilized a statistical approach to resolve anaphoric ambiguity. Specifically, they investigated whether a term was frequently referred by a pronoun in place of the subject or object of a verb. Paul et al. [99] demonstrated the corpus-based approach that utilizes the decision tree trained on a corpus annotated with the frequency information of co-referential and non-referential pairs. Ng and Cardie [100] used C4.5 decision tree algorithm [101] coupled with extra-linguistic changes and additional feature set including string match feature. Iida et al. [102] developed Search-Then-Classification model that searches most likely candidate antecedent and then classifies them using Support Vector Machine (SVM)-based anaphoricity classification model. Yang et al. [103], [104] applied ML approaches to identify nocuous ambiguities in requirements documents and reported high recall with a consistent improvement on baseline precision subject to some ambiguity tolerance levels. Broscheit et al. [105] developed the multilingual anaphoric ambiguity resolution system, named BART, that relies on entropy-based classifier for pairs of mentions. Uryupina [106] developed anaphoric ambiguity resolution system, named Corry, that relies on two SVM classifiers. Specifically, one of the classifiers determines whether two given mentions are co-referent or not, and the other classifier determines whether a given mention is anaphoric or discourse new. These two systems demonstrated good result in the SemEval-2010 task on Coreference Resolution in Multiple Languages [107]. Similarly, ML-based approaches were also prevalent at CoNLL-2011 shared task [108] as well including that of Chang et al. [109], Bjrkelund and Nugues [110], and dos Santos and Carvalho [111]. Zelaia et al. [112] utilized Singular Value Decomposition (SVD) to reduce dimensional vector space of linguistic features and applied multi-classifier system that classifies mention-pairs in the reduced dimensional vector space. Unlike the above approaches, some of the approaches are designed to utilize the advantage of both ML model and traditional rule/heuristic-based system. For example, Lee et al. [113] developed anaphoric ambiguity resolution system that combines the modern ML models and modular deterministic rule/heuristic-based system and reported the improvement over purely ML-based approaches.

Some of the existing approaches tried incorporating semantic background knowledge or world knowledge to complement the other anaphoric ambiguity resolution methods. For example, Ponzetto and Strube [114] presented the extension of a ML-based ambiguity resolution system that utilizes features extracted from WordNet [53] and Wikipedia. They also used semantic roles derived with SVM-based parser named ASSERT [115]. Bryl et al. [116] presented the approach that links the terms in text to DBpedia [117] and then links further to other resources such as YAGO [118]. They used kernel-based approach [119] as the learning algorithm for anaphoric ambiguity resolution system. Uryupina et al. [120] extracted features from the publicly available web knowledge bases, Wikipedia and YAGO [118], to leverage semantic information for improving anaphoric ambiguity resolution. Rahman and Ng [121] utilized the semantic features extracted from YAGO [118] and FrameNet [122] and reported the improved performance of mention-pair model and cluster-ranking model.

### 3.2.2 Coordination Ambiguity Resolution

Coordination ambiguity occurs when a term modifies coordinated terms or multiple coordinated terms modify a term. The sentence “slot widths and radii should conform to those of cutters” is an example of the manufacturing text with coordination ambiguity. Specifically, the expression “slot widths and radii” is ambiguous if the “slot” modifies either only the adjacent term “widths” or both of “widths” and “radii”. In this section, various approaches for the resolution of coordination ambiguity are reviewed.

First, similarity-based approaches have been developed to resolve coordination ambiguity. The approach measures the semantic similarity between conjunctions based on semantic tag or thesaurus and derives the correct coordination. For example, Kurohashi and Nagao [123] developed the coordination ambiguity resolution framework for Japanese text. They utilized the semantic similarity between the terms derived from the agreement of part-of-speech/character and the hierarchical relation defined in a Japanese thesaurus.

Similarly, Resnik [124] developed the approach that measures the semantic similarity between two words using WordNet [53] and demonstrated the use of the semantic similarity to resolve coordination ambiguity.

As for anaphoric ambiguity resolution, corpus-based statistical approaches have been studied. For example, Goldberg [125] developed an unsupervised statistical model for determining the attachment of ambiguous coordinate phrases. Goldberg demonstrated 72% of accuracy from Wall Street Journal text. Nakov and Hearst [126] also utilized n-gram statistics along with surface features and paraphrases extracted from very large corpus from the web. They reported improved result due to the larger size of the corpus. Chantree et al. [127] developed the coordination ambiguity resolution approach that relies on the word distribution information from a generic corpus. Specifically, they measured the relative frequency of the coordination, the distributional similarity of the coordinated words, and the collocation frequency between the coordinated words and their modifiers for the ambiguity resolution of coordination ambiguity. Ogren [128] developed the approach that utilizes a language model trained from the articles of PubMed Central [1]. Specifically, the approach first generates multiple candidate sentences by separating a sentence with a coordination and then derives the sentence probability to determine the best coordination structure. Hanamoto et al. [129] developed the dual composition method that utilizes both Head-driven Phrase Structure Grammar (HPSG) parsing [130] and statistical analysis of coordinate structure with alignment-based local features including surface word, part-of-speech, suffix and prefix,

[1] `http://www.ncbi.nlm.nih.gov/pmc/about/ftp.html`

and their combinations.

Linguistic-based approaches have been prevalent for coordination ambiguity resolution. The approaches utilize part-of-speech (POS) tagging, shallow parsing information (e.g. phrase), and deep parsing information (e.g. full parsing tree) to derive the linguistic information and apply rule-based matching to resolve coordination ambiguity. The methods have shown the good result as ML techniques for parsing advance. For example, Agarwal and Boggess [131] developed the algorithm that identifies the correct coordination of phrases utilizing POS tagging and shallow parsing information. Specifically, they tagged a text with part-of-speech/semantic tags and matched the tags to the head words with post-conjunction phrases to find correct coordination. Although the algorithm is simple and effective, its performance is limited due to the lack of considering the ambiguities from modifier attachment. Okumura and Muraki [132] presented English coordinate structure analysis model, named balance matching, that utilizes the symmetric patterns of parallelism. Specifically, they tagged the words with a phrase/word/morphological features and matched to the features of coordinated noun phrases. Then they derived the correct interpretation by choosing the coordination that maximizes the similarity between the feature sets. Rus et al. [133] performed POS tagging and Named Entity (NE) detection based on WordNet [53], and then applied several heuristics to bracket coordinated nouns. Nilsson et al. [134] developed the transformation of representation for dependency parsing and demonstrated the improved resolution of coordination ambiguity. Specifically, they transformed the representation of Prague Dependency Treebank (PDT) [135] from semantically oriented Prague Style (PS) to syntactically oriented Mel'čuk Style (MS) and reported that the error reduced due to the use of the different representation. Hogan [136] developed the lexicalized history-based parsing model to improve coordination ambiguity resolution. They also incorporated the detection of inconsistent noun phrase annotation in Wall Street Journal text to reduce the noise in coordination dataset. Hara et al. [137] developed a grammar defining a coordination tree and used the grammar to ensure the consistency of coordination in a sentence. They also utilized alignment-based local features to derive all the possible coordination trees and chose the coordination tree with the highest score with the perceptron trained on a corpus. Yang et al. [138] demonstrated the identification of nocuous coordination in requirements documents. They first performed POS tagging, shallow parsing, and word co-occurrence/distribution investigation, and then applied various ML algorithms, provided by WEKA package [2], along with several heuristics.

### 3.2.3 Challenges in Ambiguity Resolution of Manufacturing Text

To the best of knowledge, ambiguity resolution of manufacturing text has not been studied in manufacturing field. In addition, even though ambiguity resolution has been actively studied in NLP community, the previous chapter implies that the existing approaches cannot reliably address the ambiguities in a manufacturing text due to the lack of considering manufacturing context. Based on the literature review, existing approaches are not able to incorporate manufacturing context due to the following challenges.

[2] http://www.cs.waikato.ac.nz/~ml/index.html

1. The lack of domain-specific training corpus: While the prevalent ML-based and statistical approaches rely on training corpus to capture context, there is no standard, preferably annotated, corpus covering manufacturing context. Therefore, the approaches are not applicable to manufacturing text.

2. The lack of domain-specific knowledge bases: While some of the existing approaches incorporate background knowledge in knowledge bases [114], [116], [120], [121], the knowledge bases do not cover highly-domain specific context required for ambiguity resolution of manufacturing text. Therefore, the approaches are also not applicable to manufacturing text.

Thus, there is a necessity of a mechanism to incorporate manufacturing domain context in ambiguity resolution. To address the challenge, this dissertation demonstrates the ambiguity resolution method that utilizes the manufacturing domain ontology as the mechanism to incorporate domain context. Specifically, the method utilizes standard NLP techniques to analyze a manufacturing text and complements the NLP results with the heuristics-based approach founded on the domain context provided by the manufacturing domain ontology.

## 3.3 Problem Definition

Figure 3.1 shows the motivating example of domain-specific ambiguity resolution. The input is a manufacturing text, shown at the top of Figure 3.1, with the ambiguous phrase underlined. Specifically, as the term “slot” modifies the coordinated term “widths and radii”, it is ambiguous if the “slot” modifies

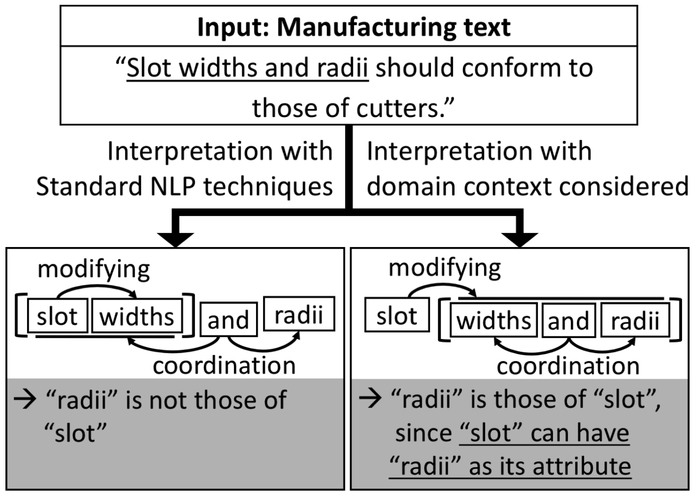



Figure 3.1: The motivating example of domain-specific ambiguity resolution is shown. The input text contains the ambiguity involved with the phrase "Slot widths and radii" since the "slot" can be regarded as modifying either only "widths" or both "widths and radii". Resolving the ambiguity requires the consideration of domain context that "slot can have radii as its attribute".

either only the adjacent term "widths" or both of "widths" and "radii". That is the reason standard NLP techniques derived the former interpretation, as shown at the bottom-left of Figure 3.1, which is not the desired interpretation. On the other hand, a domain expert can easily derive the desired interpretation based on the domain context that a "slot" can have both "widths" and "radii" as its attributes, as shown at the bottom-right of Figure 3.1. The main function of the domain-specific ambiguity resolution method is utilizing such domain context to complement standard NLP techniques so that manufacturing text with domain-specific ambiguities can be properly interpreted. Then the extraction of the desired formal rules becomes feasible from the manufacturing texts with domain-specific ambiguities. The following requirements should be addressed to achieve the proposed ambiguity resolution method.

**Ability to identify unresolved ambiguity in text** It is important to identify the existence of an unresolved ambiguity in a manufacturing text. In the motivating example (Figure 3.1), the ambiguity involved with the coordinated phrase "Slot widths and radii" should be identified when it is not correctly interpreted with standard NLP techniques.

**Ability to consider manufacturing domain context to resolve ambiguity** This entails the development of heuristics to resolve ambiguities by considering manufacturing domain context, which is the knowledge not directly written in the text but presents in the manufacturing domain ontology. In particular, some of this knowledge might be implicit, e.g., a "width" or "radius" cannot exist as an independent entity and is always associated with a manufacturing feature such as a "slot". In the motivating example (Figure 3.1), the context that "widths and radii are the possible attributes of slot and cutter" should be considered to resolve the ambiguity.

## 3.4 Overview of Ambiguity Resolution Method

Figure 3.2 shows the overview of the ambiguity resolution method. From the input text with an unresolved ambiguity, the method is designed to acquire manufacturing concepts and relations with the ambiguity resolved. The entire process is divided into three steps: the extraction of manufacturing concepts and relations, the identification of an unresolved ambiguity, and the resolution of the ambiguity.

**Extract manufacturing concepts and relations** From the input text, natural language terms and grammatical dependencies are identified by utiliz-

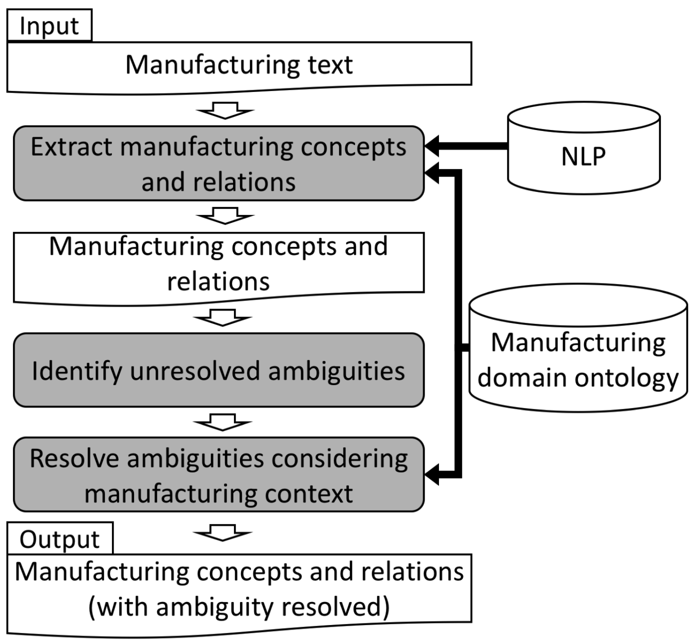



Figure 3.2: The overview of the ambiguity resolution method is shown. The block arrows indicate the sequence of execution, and the solid line arrows indicate the use of external tool or knowledge base. First, the manufacturing concepts and relations are extracted from manufacturing text as for the usual manufacturing rule extraction. Then, the unresolved ambiguities are identified and resolved considering the context provided by the manufacturing domain ontology.

ing standard NLP techniques. Then the natural language terms/grammatical dependencies are transformed to manufacturing concepts/relations defined in the manufacturing domain ontology. This step is equivalent to the "Extract natural language terms and grammatical dependencies" and "Transform to manufacturing concepts and relations" steps performed by the rule extraction framework (Figure 2.2).

**Identify unresolved ambiguities** The extracted manufacturing concepts and relations may contain unsolved ambiguities, which are domain-specific ambiguities, not addressed by standard NLP techniques. In this step, the existence of such ambiguities is identified. Two specific types of ambiguities are considered: coordination ambiguity and anaphoric ambiguity.

**Resolve ambiguities considering domain context** When the existence of unresolved ambiguities is confirmed, the ambiguities are resolved based on the domain context provided by the manufacturing domain ontology. First, additional manufacturing concepts and/or relations are added to consider all the possible interpretation of the unresolved ambiguities. Then some of the additional manufacturing concepts and relations are pruned if they are not semantically valid. The resultant manufacturing concepts and relations represent the most probable interpretation of the original manufacturing text, and the desired formal rule can be extracted based on the interpretation.

## 3.5 Extract Manufacturing Concepts and Relations

The objective of this step is analyzing manufacturing text to extract the manufacturing concepts and relations defined in the manufacturing domain ontology. First, standard NLP techniques are utilized to identify natural language terms and grammatical dependencies along with co-reference, from the input text. Figure 3.3 shows the natural language terms and grammatical dependencies derived from the motivating example. Each node (e.g. "slot", "widths", "and"...) corresponds to each term of the input text, and each connection between the nodes (e.g. "compound", "cc", "conj", "coref"...) represents the grammatical dependency or co-reference. It should be noted that co-reference resolution, i.e., anaphoric ambiguity resolution performed by standard NLP techniques, is effective only when manufacturing domain context is not involved. For example, as shown in Figure 3.3, standard NLP techniques were not able to derive the co-reference involving the pronoun "those", and that's the reason the grammatical dependency "coref" is not shown in Figure 3.3. Once the natural language terms and grammatical dependencies are identified, they are matched to manufacturing concepts or relations defined in the manufacturing domain ontology. In Figure 3.3, it is shown that the terms "slot", "widths", "radii", and "cutters" are matched to *Slot*, *hasWidth*, *hasRadius*, and *Cutter* in the manufacturing domain ontology. As stated in the previous section, this step is equivalent to the "Extract natural language terms and grammatical dependencies" and "Transform to manufacturing concepts and relations" steps performed by the formal manufacturing rule extraction framework (Figure 2.2).

## 3.6 Identify Unresolved Ambiguities

The objective of this step is identifying if the manufacturing text contains an unresolved ambiguity, i.e., domain-specific ambiguity. In this dissertation,

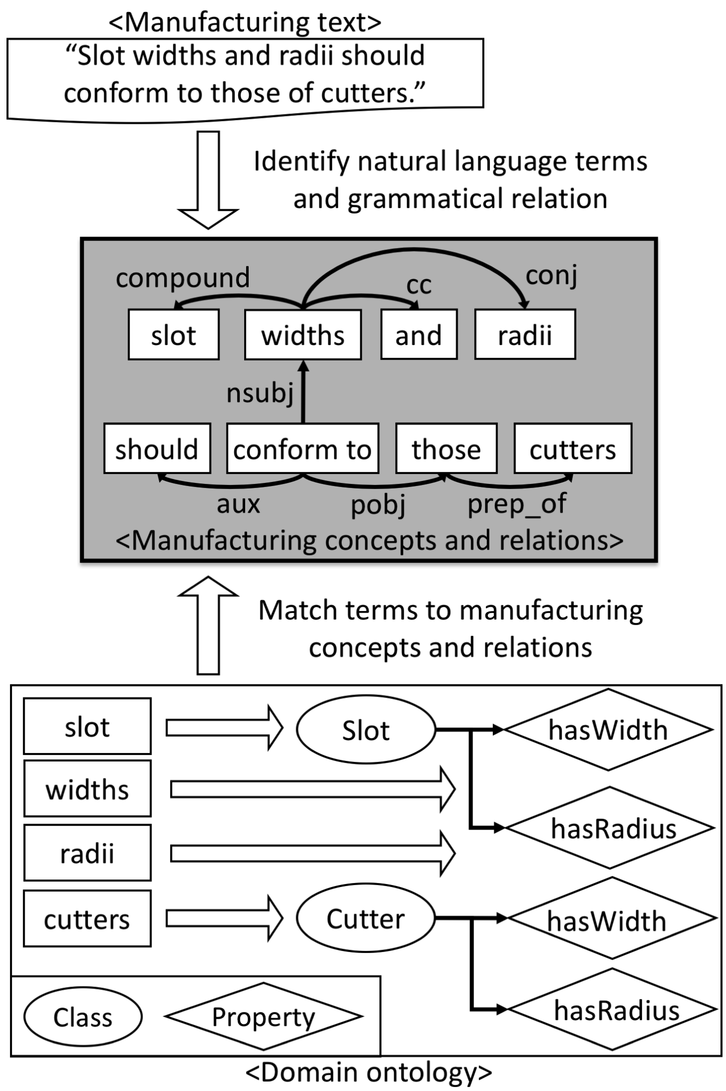



Figure 3.3: The manufacturing concepts and relations extracted from the input text of the motivating example are shown (in the shaded box). Natural language terms/grammatical relations are identified from the text, and then manufacturing-related terms are transformed to manufacturing concepts/relations(or attributes) defined in the manufacturing domain ontology.

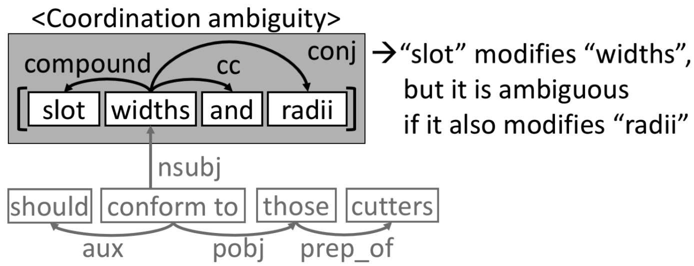



Figure 3.4: The coordination ambiguity in the motivating example (Figure 3.1) is shown. The coordination ambiguity is caused since the term "slot" modifies the coordinated terms "widths and radii". Specifically, it is ambiguous if the term "slot" modifies only the adjacent term "widths" or both "widths" and "radii".

two representative types of ambiguities are considered: coordination ambiguity and anaphoric ambiguity.

**Coordination ambiguity** Coordination ambiguity occurs 1) when a term modifies coordinated nouns or 2) coordinated terms modify a noun. To identify the existence of a coordination ambiguity, the manufacturing concepts and relations are examined if they contain either case of coordination ambiguity. Figure 3.4 shows an example of the first case of coordination ambiguity, which occurs when a term modifies coordinated terms. In the example, the coordination ambiguity occurs since the term "slot" modifies the coordinated terms "widths and radii", which are shown as the shaded region. Specifically, it is ambiguous if the term "slot" modifies either only the adjacent term "widths" or both "widths" and "radii".

On the other hand, Figure 3.5 shows the second case of the coordination ambiguity, which occurs when coordinated terms modify a term. The example can be found in the sentence "Inside and outside corners should not be sharp". In

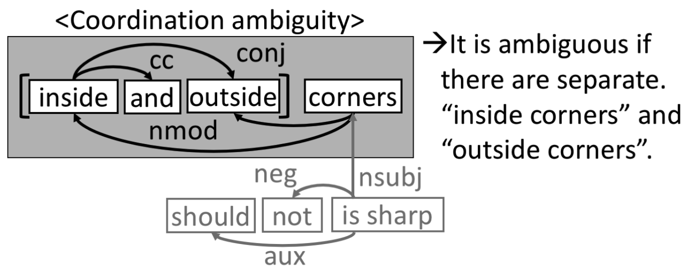



Figure 3.5: Another example of coordination ambiguity is shown. The coordination ambiguity is caused since the coordinated terms “inside and outside” modify the term “corners”. Specifically, it is ambiguous if the phrase meant separate “inside corners” and “outside corners”, or the corners are “inside” and “outside” at the same time.

the example, the coordination ambiguity occurs since the coordinated terms “inside and outside” modifies the term “corners”, which are shown as the shaded region. Specifically, it is ambiguous if the phrase meant separate “inside corners” and “outside corners”, or the corners are “inside” and “outside” at the same time.

**Anaphoric ambiguity** Anaphoric ambiguity occurs when there is a pronoun referring to something previously mentioned, and the pronoun has multiple antecedents. To identify the existence of an anaphoric ambiguity, the pronoun in the manufacturing text are examined if the standard NLP techniques provided the referent, which is the term referred by the pronoun. If the referent is not given and the pronoun has multiple antecedents, the text is regarded to contain an anaphoric ambiguity. Figure 3.6 shows the anaphoric ambiguity in the motivating example. Specifically, the anaphoric ambiguity occurs since the referents of the pronoun “those” are not provided by the standard NLP techniques, as shown in the shaded region, and there are three

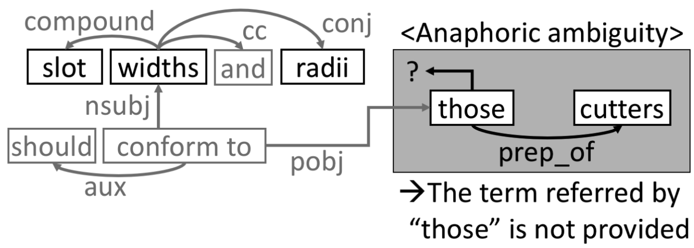



Figure 3.6: The anaphoric ambiguity in the motivating example is shown. The anaphoric ambiguity is caused since the referent of the pronoun "those" is not provided by standard NLP techniques.

possible referents, "slot", "widths", and "radii".

## 3.7 Resolve Ambiguities Considering Manufacturing Context

The objective of this step is resolving the ambiguities identified in the previous step by considering domain context provided by the manufacturing domain ontology. The resolution has two sub-steps: identifying all the possible interpretations and choosing the appropriate interpretation.

**Identifying all the possible interpretations** First, all the possible interpretations are identified by creating additional relations. For the coordination ambiguity involved with a term modifying coordinated terms, additional relations are created between the modifying term and all the coordinated terms. In other words, all the coordinated terms are assumed to be modified by the modifying term. Figure 3.7 shows that the additional relation "compound", shown as the dotted arrow, is created assuming that the term "slot" modifies both "widths" and "radii". It should be noted that the type of additional relations are "compound" when the modifying term is a noun, or "nmod" when the modifying term is an adjective.

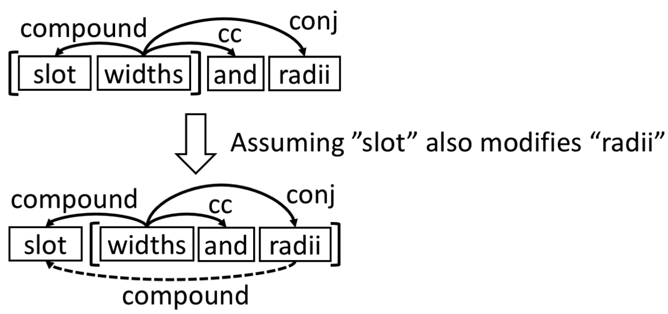



Figure 3.7: The additional relation “compound”, shown as the dotted arrow, is created to identify all the possible interpretation for the coordination ambiguity in the motivating example. Specifically, it is assumed that the term “slot” modifies both “widths” and “radii”.

On the other hand, for the coordination ambiguity involved with coordinated terms modifying a term, additional terms are created for each of the modifying terms, and additional relations are created accordingly. Figure 3.8 shows that the additional term “corner” and the additional relation “nmod”, shown as the dotted square and arrow respectively, are created. Specifically, by adding the term and relation, it is assumed that the phrase “inside and outside corners” refers to the separate “inside corners” and “outside corners”. It should be noted that the type of additional relations are “compound” when the modifying term is a noun, or “nmod” when the modifying term is an adjective.

For an anaphoric ambiguity, identifying all the possible interpretation is straightforward. In that case, additional relations “coref” are created between the unresolved pronoun and all the antecedents. Figure 3.9 shows that the additional relations “coref”, shown as the dotted arrows, are created. Specifically, by adding the relations, the pronoun “those” is assumed to refer to all the antecedents “slot”, “widths”, and “radii”.

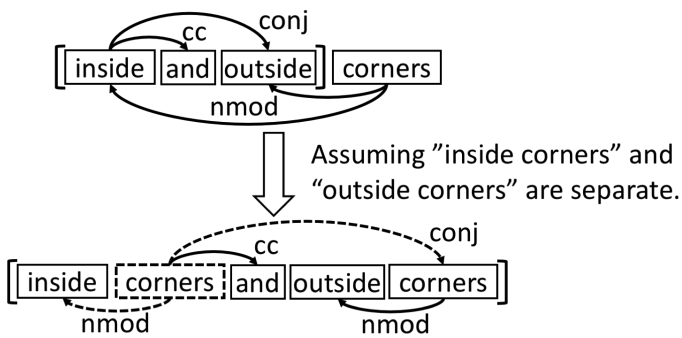



Figure 3.8: The additional term “corners” and relation “nmod”, shown as the dotted squre and arrow, are created to identify all the possible interpretation for the coordination ambiguity in the phrase “Inside and outside corners”. Specifically, it is regarded that the phrase means the separate “inside corners” and “outside corners” rather than the “corners” that are “inside and outside” at the same time.

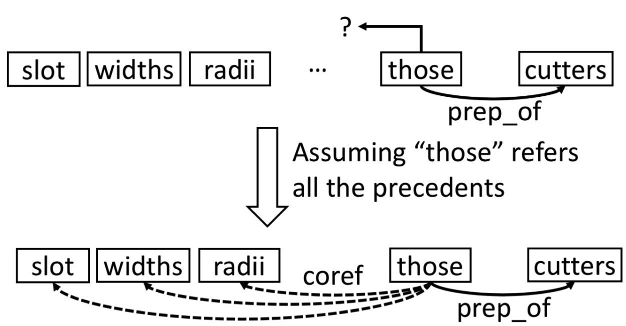



Figure 3.9: The additional relations “coref”, shown as the dotted arrows, are created to identify all the possible interpretation for the anaphoric ambiguity in the motivating example. Specifically, it is regarded that the pronoun “those” refers to all the antecedents “slot”, “widths”, and “radii”.

**Choosing the appropriate interpretation** Once the additional concepts and relations are created, the appropriate interpretation is derived by pruning semantically invalid relations and/or concepts. In pruning the additional relations, the semantic likelihood of each relation is estimated by utilizing the domain context provided by the manufacturing domain ontology. The following three criteria are used to estimate the semantic likelihood of a relation.

- The relation between two concepts is regarded semantically valid if the manufacturing domain ontology includes a semantic relation between the two involving concepts.
  e.g. The concepts *Hole* and *EntranceSurface* are regarded to be semantically related since the manufacturing domain ontology includes the semantic relations *hasEntranceSurface* between the concepts.

- Two concepts are regarded as semantically related when a concept can be the attribute of the other concept.
  e.g. The concepts *Slot* and *Width* are regarded to be semantically related since the manufacturing domain ontology defines that the concept *Slot* can have the attribute *hasWidth*.

- The relation between two concepts is regarded as semantically valid when the two concepts can form another concept as a whole.
  e.g. When the term "inside" modifies another term "corner", the modifying relation is regarded to be semantically valid since the manufacturing domain ontology defines the combined concept *InsideCorner* as a specific type of (i.e. *subclass* of) *Corner*.

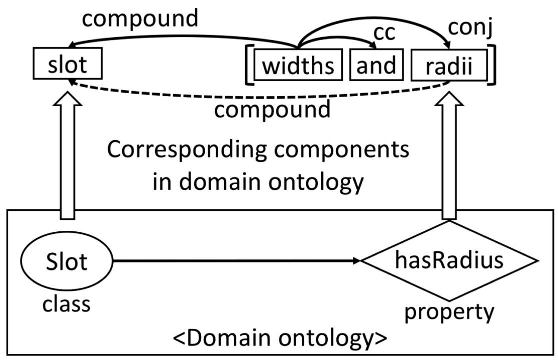



Figure 3.10: The resolution of the coordination ambiguity in the motivating example, occurred by one modifying term and coordinated nouns, is shown. Since the manufacturing concept *Slot* has the attribute *hasRadius* in the manufacturing domain ontology, the additional "compound" relation (shown as the dotted arrow) is kept.

For the coordination ambiguity involved with a term modifying multiple coordinated terms, the manufacturing domain ontology is examined if it defines the semantic relation between the manufacturing concepts related through the additional relations. Figure 3.10 shows how the semantic relation is estimated in case of the coordination ambiguity in the motivating example. Specifically, as the manufacturing domain ontology defines that the manufacturing concept *Slot* has the attribute *hasRadius*, it is considered that the term "slot" modifies the term "radii". Therefore, the additional relation "compound", shown as the dotted arrow, is kepted.

For a coordination ambiguity involved with multiple coordinated terms modifying a term, the manufacturing domain ontology is examined if it contains a separate concept that can be derived from the additional concepts and relations. Figure 3.11 shows how the process is performed. Specifically, as the manufacturing domain ontology defines the concept *InsideCorner* that can be

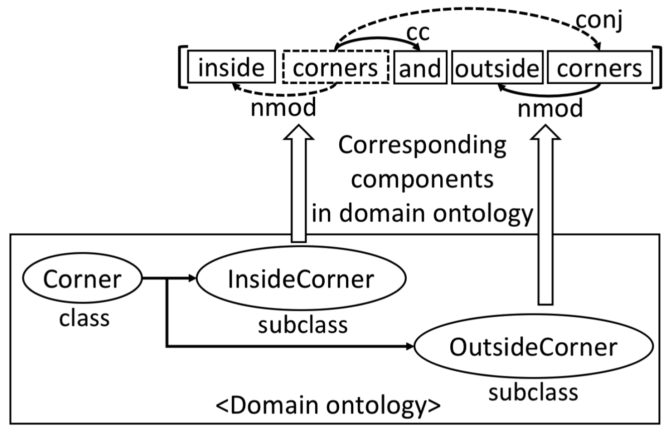



Figure 3.11: The resolution of the coordination ambiguity, occurred by coordinated modifying terms, is shown. Since there is the manufacturing concept *InsideCorner* corresponding to "inside corner", it is regarded that the phrase "inside and outside corners" meant separate "inside corners" and "outside corners".

derived by combining the modifying term "inside" and the additional term "corners", it is regarded that the phrase "inside and outside corners" meant separate "inside corners" and "outside corners". Therefore, the additional concept *Corners* and additional relation "nmod", shown as the dotted square and arrow, are kept.

On the other hand, for an anaphoric ambiguity, the manufacturing domain ontology is examined if it defines the semantic relation between the antecedents and the terms around the pronoun. Figure 3.12 shows how the semantic relation is estimated in case of the anaphoric ambiguity in the motivating example. Specifically, from the phrase "those of cutter", it is assumed that the pronoun "those" should refer to the attributes of the term "cutters". In the meanwhile, the manufacturing concept *Cutter* has the attributes *hasWidth* and *hasRadius* while it does not have *hasSlot*. As a result, the pronoun is considered only

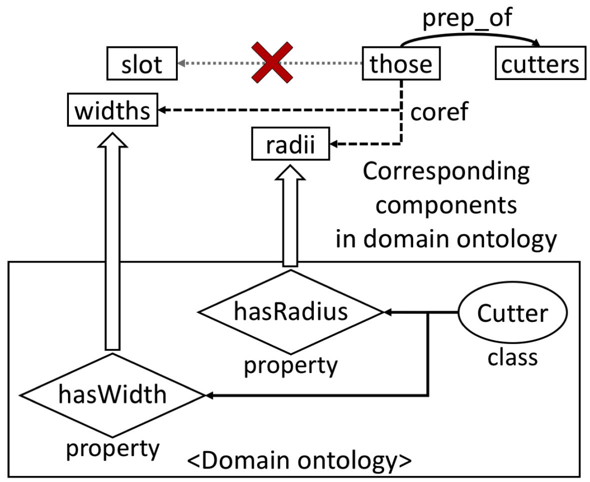



Figure 3.12: The resolution of the anaphoric ambiguity in the motivating example is shown. Since the manufacturing concept *Slot* has the attribute *hasWidths* and *hasRadius* and does not have the attribute *hasSlot*, the relation “coref” to the term “slot” is pruned, shown as the faded arrow, while the other “coref” relations are kept, shown as the solid dotted arrows.

refers to the terms “widths” and “radii”. Therefore, the additional relation “coref” to the term “slot” is pruned, shown as the faded arrow, while the other “coref” relations, shown as the solid dotted arrows, are kept.

Once this step is done, ambiguities in the manufacturing concepts/relations are resolved, and the extraction of the desired formal rules becomes feasible with the previously developed rule extraction framework. Figure 3.13 compares the formal manufacturing rules extracted from the motivating example before and after ambiguity resolution. Specifically, it is shown that the unresolved component *null* is resolved after coordination ambiguity resolution, and the other unsolved component *those* is resolved after anaphoric ambiguity resolution.

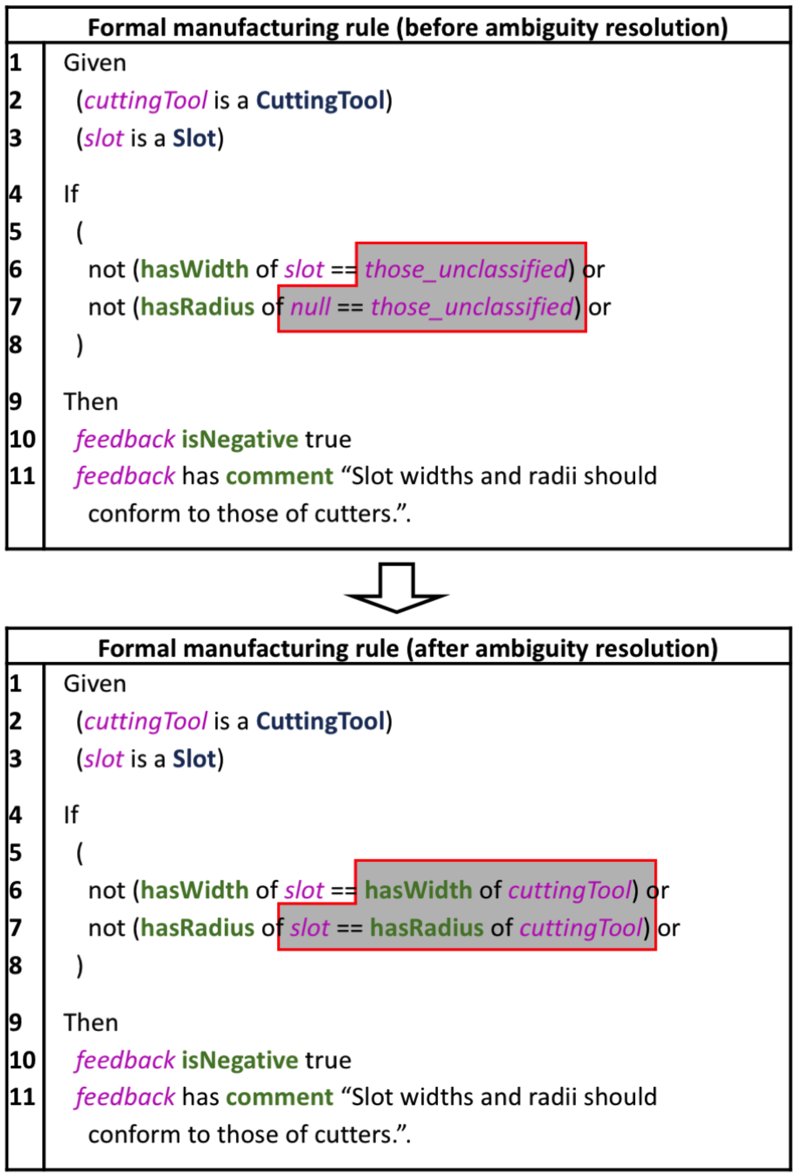



Figure 3.13: The formal manufacturing rules extracted from the motivating example, before and after ambiguity resolution, are compared. The unresolved component, *null* and *those* are addressed after ambiguity resolution

## 3.8 Implementation and Experiment

### 3.8.1 Experimental Setting

**Test scenario and dataset** As the test scenario, the rule extraction framework is extended to adopt the ambiguity resolution method. The same test and validation dataset were used for the development of the extended framework. For more details about the dataset creation and the list of sentences in the dataset, the readers are referred to Appendix A of this dissertation.

**Construction of manufacturing domain ontology** The manufacturing domain ontology used in the previous chapter does not require any modification to incorporate ambiguity resolution. It implies that the ambiguity resolution method can be applied to different types of semantics-based applications when the domain ontology provides enough semantics to perform ambiguity resolution. Figure 3.14 shows the part of the manufacturing domain ontology corresponding to the domain context relevant to the motivating example (Figure 3.1).

### 3.8.2 Implementation

To verify the effectiveness of the ambiguity resolution method, the previously developed Java application for rule extraction is extended to adopt the ambiguity resolution method. As mentioned in the previous chapter, the application utilizes NLP4J [79] and OpenNLP [80] as the NLP tools and utilizes Apache Jena Ontology API [83] for the ontology management and reasoning. Figure 3.15 shows the screenshots of the application. The interface of the application is updated to provide the ambiguity resolution processes so that the

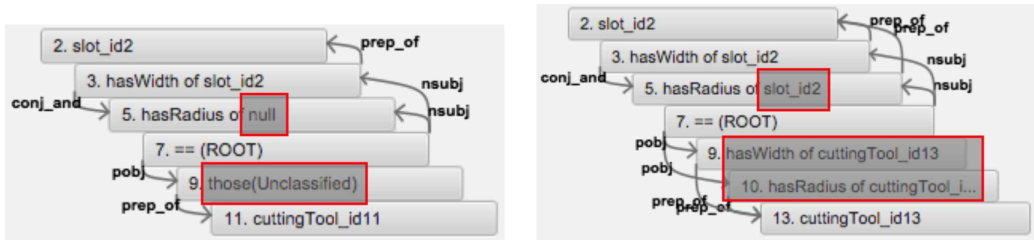
```
Slot is a type of ManufacturingFeature.
hasWidth describes {PhysicalResource, Slot, Keyway}
    with a single value of type float.
hasRadius describes {PhysicalResource, Slot, ManufacturingFeature}
    with a single value of type float.

PhysicalResource is a type of Resource.
    CuttingTool is a type of PhysicalResource.

    hasRadius describes CuttingTool with a single value of type float.
    hasWidth describes CuttingTool with a single value of type float.
```

Figure 3.14: A part of manufacturing domain ontology corresponding to the motivating example (Figure 3.1) is shown. The part specifies that the manufacturing concepts *Slot* and *CuttingTool* have the attributes *hasWidth* and *hasRadius*, which are the context relevant to resolve the ambiguities in the motivating example.

user can understand how the ambiguity is resolved. Figure 3.15 (a)–(c) shows the information provided by the updated application for the motivating example. Specifically, Figure 3.15 (a) shows the manufacturing concepts/relations before ambiguity resolution, Figure 3.15 (b) shows the manufacturing concepts/relations after ambiguity resolution, and Figure 3.15 (c) shows the extracted rule after ambiguity resolution. It is shown that the unresolved components of the manufacturing concepts/relations, which are *null* and *those* shown as the shaded regions in Figure 3.15 (a) and (b), are addressed after ambiguity resolution.

### 3.8.3 Result and Discussion

The extraction results before and after adopting the ambiguity resolution method are presented in Table 3.1. The effectiveness of the proposed ambiguity resolution method is verified by an increasing the correct rules to 70% (increased by approx. 13%). In addition, it is manually verified that all the




(a) (b)

```
Rule DesignRule2

// Performing general if-then rule extraction

given
      (design_id0 is a MechnicalDesignService)

      (abrasiveMachining_id1 is a AbrasiveMachining)
            and (abrasiveMachining_id1 is hasProcess of design_id0)
            and (hasPart of design_id0 is hasTargetPart of abrasiveMachining_id1)
      (cuttingTool_id13 is a CuttingTool)
            and (cuttingTool_id13 is hasTool of abrasiveMachining_id1)
      (slot_id2 is a Slot)
            and (slot_id2 is hasFeature of hasPart of design_id0)

if
      // Supporting relation

      // Main relation
      (
            not (hasWidth of slot_id2 == hasWidth of cuttingTool_id13) or
            not (hasRadius of slot_id2 == hasRadius of cuttingTool_id13)
      )

then
      feedback2 isNegative true
      feedback2 has comment "Slot widths and radii should conform to those of c
```

(c)

Figure 3.15: The screenshots of the updated Java application with domain-specific ambiguity resolution are shown. The updated interface provides the ambiguity resolution processes as follows: (a) manufacturing concepts/relations before ambiguity resolution, (b) manufacturing concepts/relations after ambiguity resolution, and (c) the output formal manufacturing rule with ambiguity resolved. The unresolved component of manufacturing concepts/relations are shown as the shaded regions in (a) and (b).

ambiguities in the dataset have been resolved. It should be noted that the extraction of the desired formal rule was not successful from the rest of the cases (approx. 30%), which are due to the issues such as the complex sentence structure or the inclusion of unnecessary information. In the rest of this section, the successful case and the unsuccessful case are reviewed. For the full list of the additional 17 correct formal rules obtained after adopting ambiguity resolution, the readers are referred to Appendix E of this dissertation.

Table 3.1: The result of the case study is shown. Adopting the ambiguity resolution method increases the correct rules to 70% (increased by approx. 13%).

| Total # of sentences | Correct rule obtained (before AR) | Correct rule obtained (after AR) |
| --- | --- | --- |
| 133 | 76 | 93 |

**Ambiguity is successfully resolved** In this case, the successful resolution of ambiguities in a manufacturing text is verified. Please note that the uniqueness of the representation of the rule is not considered in this dissertation, while there may be multiple representations of the manufacturing text.

Figure 3.16 shows one of the examples involved with anaphoric ambiguity resolution. The input sentence "Punches for round holes are more economical than those of square holes." contains an anaphoric ambiguity as the referent of the pronoun "those" was not provided by standard NLP techniques, shown as the shaded region. In the example, the pronoun considered only refers to the term "punches" based on the context provided by the manufacturing domain ontology. Specifically, while the relation *hasTargetFeature* between the manufacturing concept *Punch* and the other concept *Hole* justifies the semantic relatedness of the co-reference to the term “Punches”, the semantic relatedness of the other co-reference to the term “hole” was not justified by domain context. Figure 3.17 compares the formal manufacturing rules extracted from the example before and after ambiguity resolution. Specifically, it is shown that the relation between *Punch* and *Hole* is correctly captured via *hasTargetFeature* after addressing unresolved pronoun “those”.

The Figure 3.18 shows the next example, which involves coordination ambiguity occurred by coordinated modifying terms. The input sentence “Inside and outside corners of a part should not be sharp.” contains a coordination ambiguity, which involves with the phrase “inside and outside corners”, shown as the shaded region. Specifically, the phrase can imply either separate “inside corners”/“outside corners” or the corners that are “inside” and “outside” at the same time. In the example, the former interpretation is considered to be correct as the manufacturing concept *Corner* has the subclasses *InsideCorner* and *OutsideCorner*. Figure 3.19 compares the formal manufacturing rules extracted from the example before and after ambiguity resolution. Specifically, it is shown that separate *InsideCorner* and *OutsideCorner* are properly captured after ambiguity resolution.

**Ambiguity is not resolved as input text validation is required** In these cases, the extraction of the desired formal rules was not successful due to the issues of input text such as the inclusion of complex sentence structure or unnecessary information. For example, standard NLP techniques are not

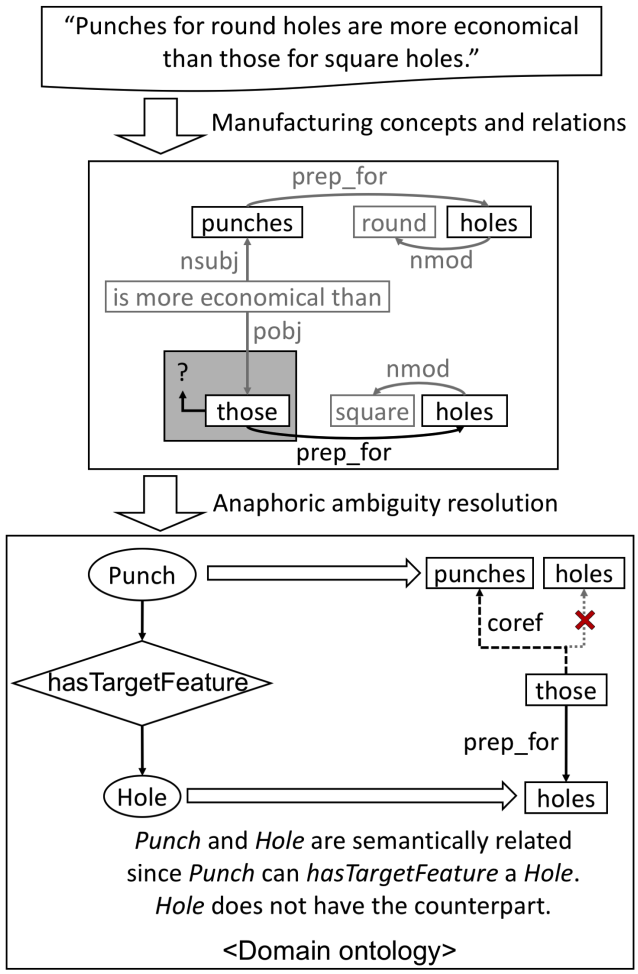



Figure 3.16: The example of successful anaphoric ambiguity resolution is shown. The sentence contains an anaphoric ambiguity as the referent of the pronoun "those" is not provided by standard NLP techniques, shown as the shaded region. The proposed method was able to identify the proper referent of the pronoun, "punches", as a *Punch* can be related with a *Hole* through *hasTargetFeature*.

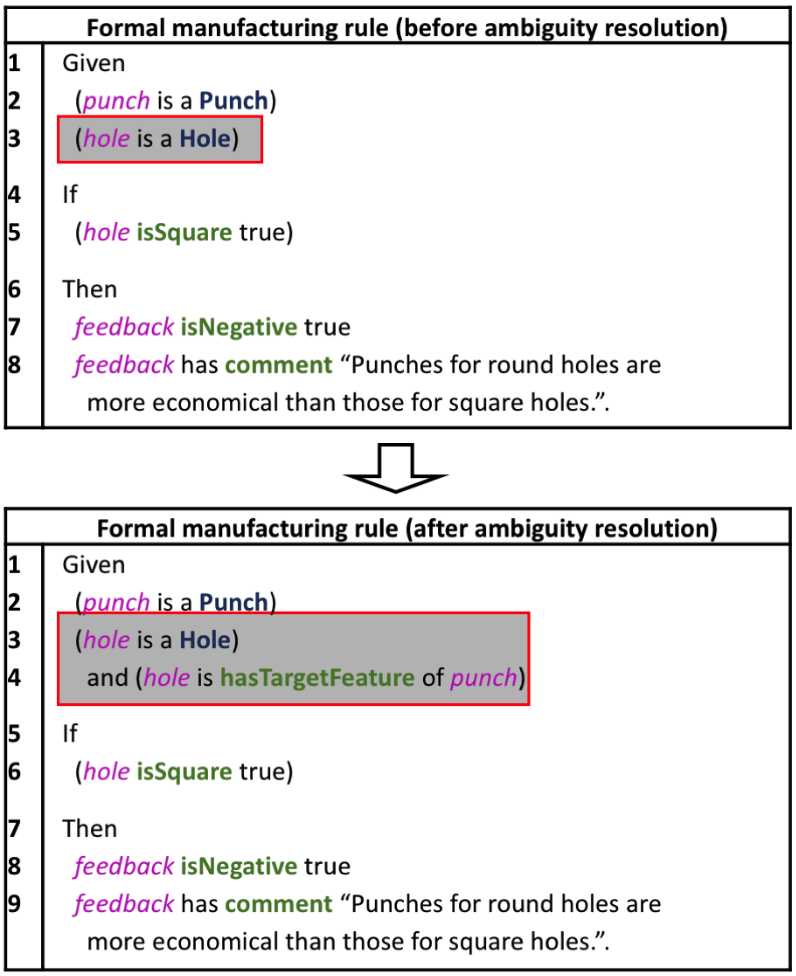



Figure 3.17: The formal rules extracted from the manufacturing text with anaphoric ambiguity, before and after ambiguity resolution, are compared. It is shown that the relation between *Punch* and *Hole* is correctly captured via *hasTargetFeature* after addressing unresolved pronoun “those”

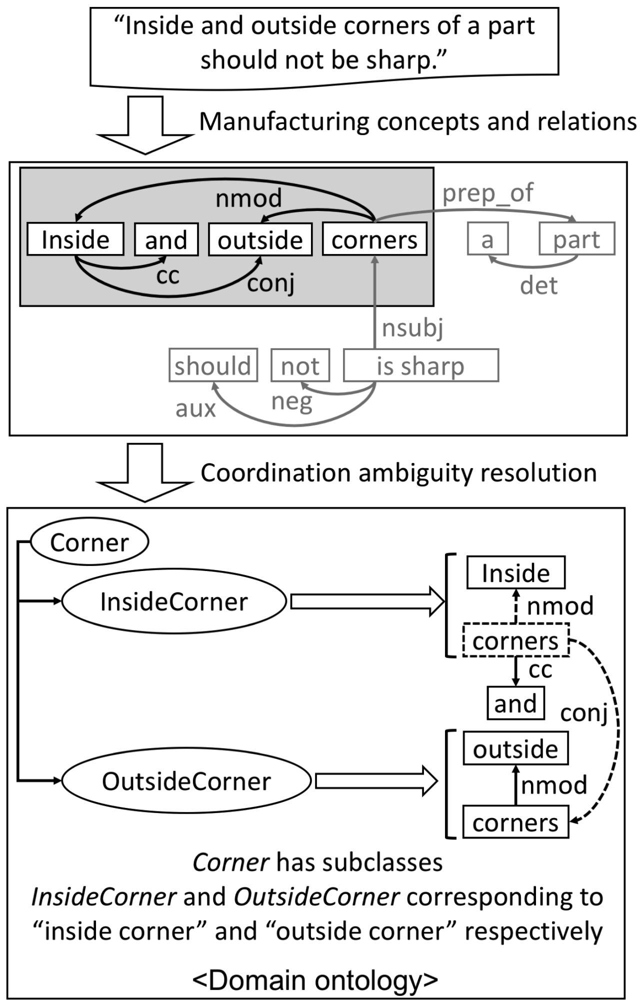



Figure 3.18: The example of successful coordination ambiguity resolution is shown. Specifically, the coordination ambiguity involves with the phrase "inside and outside corners" as the phrase can imply either separate "inside corners" and "outside corners" or "inside and outside" corners. In this example, the former interpretation is considered to be correct as the concept *Corner* has subclasses *InsideCorner* and *OutsideCorner*.

**Formal manufacturing rule (before ambiguity resolution)**

```
Given
 (outside_corner is a OutsideCorner)

If
 (outside_corner isInside true)
 (outside_corner isSharp true)

Then
 feedback isNegative true
 feedback has comment "Inside corners and outside corners
   should not be sharp.".
```

**Formal manufacturing rule (after ambiguity resolution)**

```
Given
 (inside_corner is a InsideCorner)
 (outside_corner is a OutsideCorner)

If
 (
  (inside_corner isSharp true) or
  (outside_corner isSharp true)
 )

Then
 feedback isNegative true
 feedback has comment "Inside corners and outside corners
   should not be sharp.".
```

Figure 3.19: The formal rules extracted from the manufacturing text with coordination ambiguity, before and after ambiguity resolution, are compared. It is shown that separate *InsideCorner* and *OutsideCorner* are correctly captured after ambiguity resolution

able to derive the actual subject of the sentence "It is better not to specify a blended radius on machined rails." as the subject is the dummy pronoun "it". In this dissertation, such issues are addressed by providing the relevant feedback for input text validation so that the user can modify the input text to acquire the desired formal rule. The details about the feedback generation are explained in Chapter 4. It should be noted that developing new NLP techniques to address the issues is out of scope of this dissertation.

## 3.9 Summary

In this chapter, the ambiguity resolution method is developed to extend formal manufacturing rule extraction framework. Specifically, standard NLP techniques are complemented with the domain context provided by the manufacturing domain ontology to address domain-specific ambiguity, which is the ambiguity involved with domain-specific context. By addressing domain-specific ambiguity, the desired formal manufacturing rule can be acquired even from the manufacturing text with coordination ambiguity and/or anaphoric ambiguity. The effectiveness of the ambiguity resolution method is demonstrated by significantly improving rule extraction results.

However, the experimental result showed the cases that need input text validation due to the the issues such as the inclusion of complex sentence structure or unnecessary information. It should be noted that the issues cannot be addressed by standard NLP techniques coupled with domain context, and improving the NLP tools are out of the scope of this dissertation. In this context, the next chapter discusses the development of the feedback generation method for input text validation. Specifically, the method provides the relevant feedback to the user to modify the input text accordingly so that correct rule can be acquired from the modified input text.

# CHAPTER 4: FEEDBACK GENERATION FOR INPUT TEXT VALIDATION

## 4.1 Motivation

The previous chapter demonstrated the ambiguity resolution method to extend the formal manufacturing rule extraction framework. However, the extraction of the desired formal rule is not successful for the cases that need input text validation, due to the issues such as the inclusion of complex sentence structure and unnecessary information. For example, when a formal manufacturing rule is extracted from the sentence “sharp corners should be avoided because they interfere with metal flow”, the extracted rule also contains the information corresponding to “they interfere with metal flow”, which is unnecessary in the extracted rule. To address such cases, this chapter proposes the feedback generation method for input text validation. Specifically, the method identifies the cases that need input text validation and provides the relevant feedback to the user so that the user can modify the input text to acquire correct rules.

While feedback generation has been actively studied for Intelligent Tutoring System (ITS), it has not been studied for the extraction of manufacturing rule. Therefore, there is a need to apply one of the existing methods to incorporate manufacturing context for the feedback generation for formal manufacturing rule extraction. In this context, this chapter presents the development of the feedback generation method that utilizes one of the existing feedback generation techniques, Constraint-based Modeling (CBM), coupled with standard NLP techniques and the manufacturing domain ontology. In addition, the effectiveness of the method is demonstrated by extending the rule extraction framework and providing relevant feedbacks to address the cases that require input text validation.

The remainder of this chapter is organized as follows. First, Section 4.2 reviews the existing works relevant to feedback generation. Then Section 4.3 defines the main problem of this chapter, and Section 4.4 presents the overview of the developed method. The detailed processes for feedback generation are presented in Section 4.5 and Section 4.6. Lastly, the implementation and experimental result are shown in Section 4.7, and Section 4.8 summarizes this chapter.

## 4.2 Relevant Work

In this section, the previous efforts on feedback generation are reviewed, and the reason for adopting CBM for the feedback generation method is followed. It should be noted that feedback generation for computer programming tutoring was mainly reviewed due to the availability of extensive review paper [139]. For additional details about feedback generation feature of Intelligent Tutoring System (ITS), the readers are referred to the literature [139].

### 4.2.1 Feedback Generation Approach

The categorization of feedback generation approaches is performed by Keuning et al. [139], and this dissertation sticks to their categorization. In the rest of this section, the brief review of the feedback generation approaches is followed.

The first approach is Model Tracing (MT) that creates feedbacks based on the procedure performed by a user. Specifically, each step performed by a user is compared with the desired procedure, and the hint or solution is provided when the user failed to take the desired step. The LISP tutor [140] is the example of an MT system that utilizes a set of production rules to compare each step performed by a user and the desired procedure. On the other hand, Constraint-based Modeling (CBM) [141] creates feedback only based on the user's final answer. Specifically, the method defines the constraints that should be satisfied or violated by the solution and checks if the user's answer also satisfies or violates the same constraints. As each constraint has feedback attached, relevant feedback is provided when the constraint is violated. The examples of CBM-based systems include the system named DB-suite [142] and INCOM [143]. DB-suite [142] has a CBM-based tutor that utilizes a knowledge base of constraints. Using the constraints, the system identifies if the user's answer is fully or partially equivalent to the solution. The system then provides the feedback defined in the triggered constraints. Similarly, INCOM [143] utilizes a semantic table, which includes the constraints with different weights, and transformation rules to identify the intention of a user and generate feedbacks.

While the above-mentioned methods are generic methods, some of the feedback generation methods are specifically designed to provide feedbacks for computer programming. The methods include the following: Basic Static code Analysis (BSA), Program Transformation (PT), and Intention-based Diagnosis (IBD). First, BSA method analyzes source code, byte code, and/or comments to identify the mistakes in them. Specifically, the method identifies if necessary concepts are present and unnecessary concepts are absent. Common syntactic mistakes are also identified and the appropriate feedbacks are provided to the user. The examples of BSA-based systems include Generic Automated Marking Environment (GAME-2+) [144] and InSTEP [145]. GAME-2+ [144] identifies the meaningfulness of the comments in the user's source code by examining the ratio of nouns and conjunctions compared to the total word count. InSTEP [145] identifies common errors in a source code, such as using = in place of ==. Similar to BSA, PT method also analyzes the source code submitted by a user. However, the method takes an additional step before analyzing the code. Specifically, a PT-based method first transforms the code into the canonical form, and the canonical code is analyzed to provide the relevant feedback. The examples of PT based systems include that of Xu and Chee [146] and INTELLITUTOR [147]. Xu and Chee [146] developed a PT-based system that compares the semantics of a user's source code and the canonical code. Specifically, their system identifies 13 semantic-preserving variations (SPVs) to compare a user's source code and the canonical code at the semantic level and provides feedback based on the SPVs. INTELLITUTOR [147] uses the abstract language AL as the canonical form so that C and Pascal code can be transformed to AL for the diagnosis without language-specific features. Unlike BSA and PT, IBD method focuses on the intention of a user. Specifically, the method matches the answer with the knowledge base of programming goals, plans, or (buggy) rules and tries to find if the user's intention (i.e., algorithm) is correct. Proper feedbacks are given based on the comparison of the desired intention and the user's intention. For example, the IBD-based system PROUST [148] utilizes a knowledge base of programming plans to recognize the intention of the user's source code. While the above-mentioned methods provide useful mechanisms to generate relevant feedbacks for the education of computer programming, the methods are not applicable to different areas due to the different nature of the problem.

Other recently introduced methods include Artificial Intelligence (AI)-based and Machine Learning (ML)-based approaches. For example, Lane and Van-Lehn [149] developed a dialogue-based intelligent tutoring system ProPL that mimics the conversation between a human tutor a user. Macnish developed the system datlab [150] that utilizes an Artificial Neural Network (ANN) to classify the error of a user and generate relevant feedbacks. Even though ANN-based approaches are potentially powerful mechanisms to generate proper feedback, they are not applicable to revise manufacturing text due to the lack of training resource. Ferreira and Atkinson [151] developed a feedback generation system for foreign language tutoring using a simple decision tree to choose the type of feedback to be given. Gutierrez and Atkinson [152] utilized the combination of two ML methods, SVM and CRF, trained from a set of the actual user-tutor interactions to generate adaptive feedback for virtual foreign language tutor. NLP has been frequently used for ITS especially for foreign language tutor. Nagata [153] developed a system named BANZAI that utilizes NLP to analyze the Japanese input of a user and generates the relevant feedback if the input has a grammatical deficiency. Dzikovska et al. [154] developed a tutoring system for basic electricity and electronics, called BEETLE II. Their system also utilizes NLP coupled with an ontology to analyze the user's answer and find out if some of the required concepts are missing in the answer. The system can generate dynamic feedbacks based on the result of the analysis.

#### 4.2.2 Feedback Generation for Formal Manufacturing Rule Extraction

As mentioned, automated feedback generation has not been studied in manufacturing field for formal rule extraction. Thus, there is a necessity of adopting an existing approach to perform feedback generation for manufacturing rule extraction. In this dissertation Constraint-based Modeling (CBM) is adopted for the following reasons.

1. The methodology of CBM is applicable regardless of the domain of interest, while some of the other approaches (BSA, PT, IBD) are specifically designed for the feedback for computer programming.

2. CBM does not require standard training resources as recent AI-based or ML-based approaches do.

3. CBM can be used without inspecting the procedure of the action performed by a user while MT cannot.

4. The cases that need input text validation, such as the inclusion of subordinate clause with unnecessary information, can be modeled as the constraints that CBM relies on.

To adopt CBM to develop the feedback generation method for formal manufacturing rule extraction, previously extracted manufacturing rules are analyzed to model the constraints that identify the cases that require input text validation. When a constraint is triggered, the relevant feedback is provided so that the user can modify the input text to extract the correct manufacturing rule.

## 4.3 Problem Definition

Figure 4.1 shows the motivating example of feedback generation. The input is the manufacturing text, shown at the top-left of Figure 4.1, which needs input text validation for the extraction of the desired formal rule. Specifically, the underlined subordinate clause "because they interfere with metal flow" represents the side effect of the design, which is unnecessary in the extracted rule to identify the impediment for manufacturing. The inclusion of unnecessary information is not desired as it makes the rule redundant and not exactly correct. It should be noted that identifying unnecessary information is not feasible with standard NLP techniques even coupled with domain context. To address the challenge, the main function of the feedback generation method is to identify such cases and provide relevant feedback to the user. Based on the feedback, the user can modify the input text and acquire the correct rule with the rule extraction framework. For example, Figure 4.1 shows that the feedback suggests the user check the subordinate clause and enclose the clause with parenthesis when the information is unnecessary. As the suggested option does not require the user to learn formal structured languages such as the Web Ontology Language (OWL) [10] or Semantic Application Design Language (SADL) [11], the user can easily mark the unnecessary information and utilize the rule extraction framework to acquire the correct rule. Developing

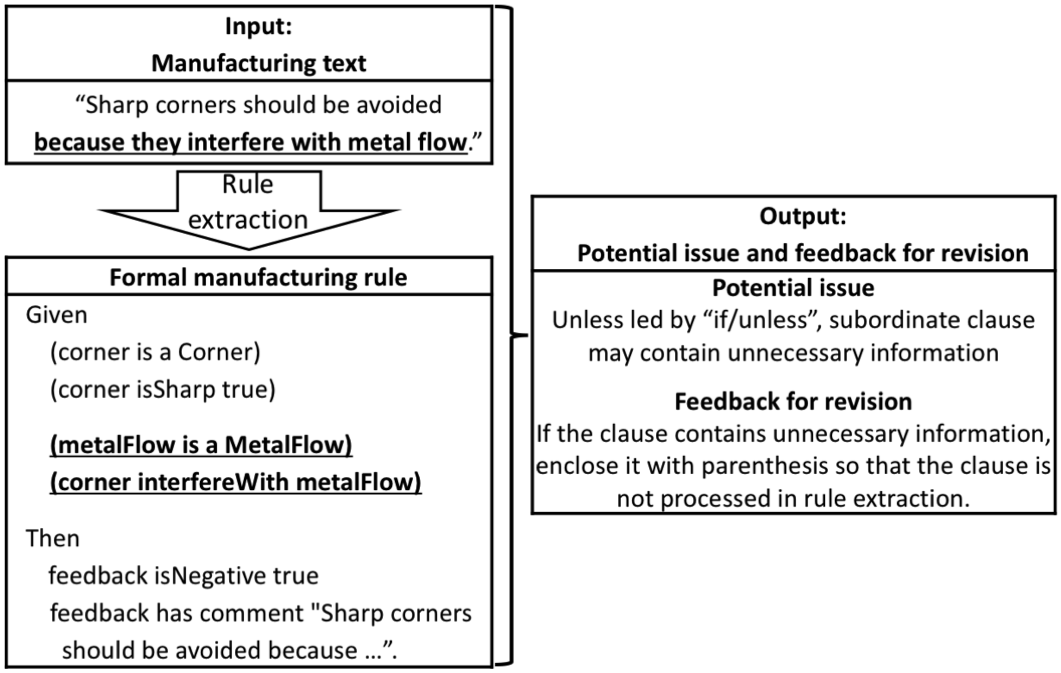



Figure 4.1: The motivating example of feedback generation is shown. The input text, shown at the top-left, needs modification as it contains unnecessary information, which is underlined. To address the case, it is desired to generate the relevant feedback for the input text validation, shown at the right, so that the user can modify the input text to acquire correct rules.

the proposed feedback generation method requires addressing the following requirements.

**Ability to identify the cases that need input text validation** The cases that need input text validation should be identified. The issues of input text include: 1) the inclusion of unnecessary information, 2) the existence of the semantically invalid semantic relation between concepts, and 3) the input sentence without explicit subject. For example, Figure 4.1 shows that the subordinate clause "because they interfere with metal flow" needs not to be considered by the rule extraction framework.

**Ability to generate the relevant feedback for input text validation** Once the cases that need input text validation are identified, relevant feedback should be generated to provide helpful information including: 1) the part of input text related to the issue, 2) the detailed explanation of the issue, and 3) the options to address the issue. It should be noted that the user should be able to address the issue without learning formal structured languages. For example, Figure 4.1 shows that the feedback suggests the user enclose the unnecessary part with parenthesis so that the unnecessary information is ignored in the rule extraction process.

## 4.4 Overview of Feedback Generation Method

Figure 4.2 shows the overview of the feedback generation method. From the manufacturing text which needs validation, relevant feedback is generated utilizing constraints to help the user address the issue. The entire feedback generation process is divided into the following two steps: the extraction of manufacturing concepts/relations and the identification of triggered constraints and feedback generation.

**Extracting manufacturing concepts and relations** In this step, manufacturing concepts and relations are derived. First, natural language terms and grammatical dependencies are identified utilizing NLP tools. Then the natural language terms/grammatical dependencies are matched to manufacturing concepts/relations defined in the manufacturing domain ontology. This step is important since the manufacturing concepts and relations are examined to identify which constraints are triggered. This step is equivalent to the

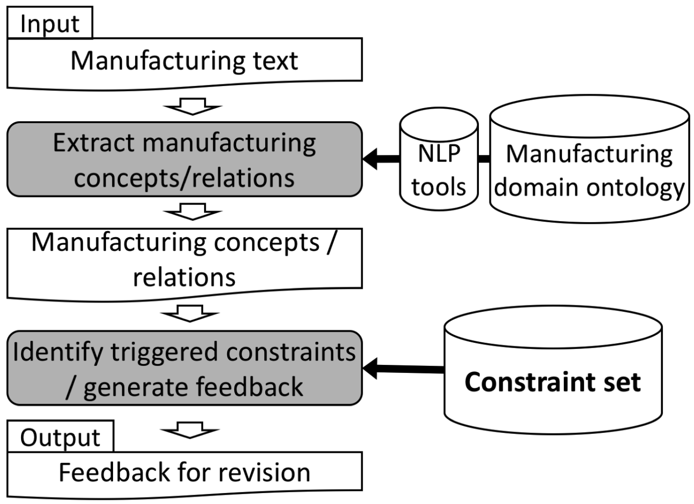



Figure 4.2: The overview of the feedback generation method is shown. First, Natural Language Processing (NLP) and the manufacturing domain ontology are utilized to extract manufacturing concepts/relations. Then the constraint set is used to identify the cases that need input text validation and generate relevant feedbacks to help the user modify the input text.

"Extract natural language terms and grammatical dependencies" and "Transform to manufacturing concepts and relations" steps performed by the formal manufacturing rule extraction framework (Figure 2.2).

**Identifying triggered constraints and generate feedbacks** In this step, relevant feedback for input text validation is provided so that the user can address the issues. To identify the cases that need input text validation, which are the candidates for feedback generation, a set of constraints are designed. Specifically, if the manufacturing concepts and relations trigger some of the constraints, the feedback attached to the constraints is provided. Based on the feedback, the user can modify the input text to acquire correct rule.

## 4.5 Extract Manufacturing Concepts and Relations

This step is the preliminary step that translates a manufacturing text into the manufacturing concepts and relations, which are examined to find out if the input text needs validation. The overall process of deriving manufacturing concepts and relations is shown in Figure 4.3. First, standard NLP techniques are utilized to analyze the manufacturing text and identify natural language terms/grammatical dependencies including co-references. Specifically, each node (e.g. "sharp", "corner", "should" ...) corresponds to each term of the input text, and each connection between the nodes (e.g. "nsubj", "aux", "dobj", "coref" ...) represents the grammatical dependency or co-reference. Then the natural language terms/grammatical dependencies are transformed to the manufacturing concepts/relations defined in the manufacturing domain ontology. Specifically, the terms "corner", "sharp", and "metal flow" are matched to *Corner*, *isSharp*, and *MetalFlow* in the manufacturing domain ontology. As mentioned in the previous section, this step is equivalent to the "Extract natural language terms and grammatical dependencies" and "Transform to manufacturing concepts and relations" steps performed by the formal manufacturing rule extraction framework (Figure 2.2).

## 4.6 Identify Triggered Constraints and Generate Feedbacks

In this step, a set of constraints are utilized to examine the manufacturing concepts/relations so that possible issues of the input text can be identified and relevant feedback can be provided. In the rest of this section, constraint check mechanism and the categories of the constraints are presented.

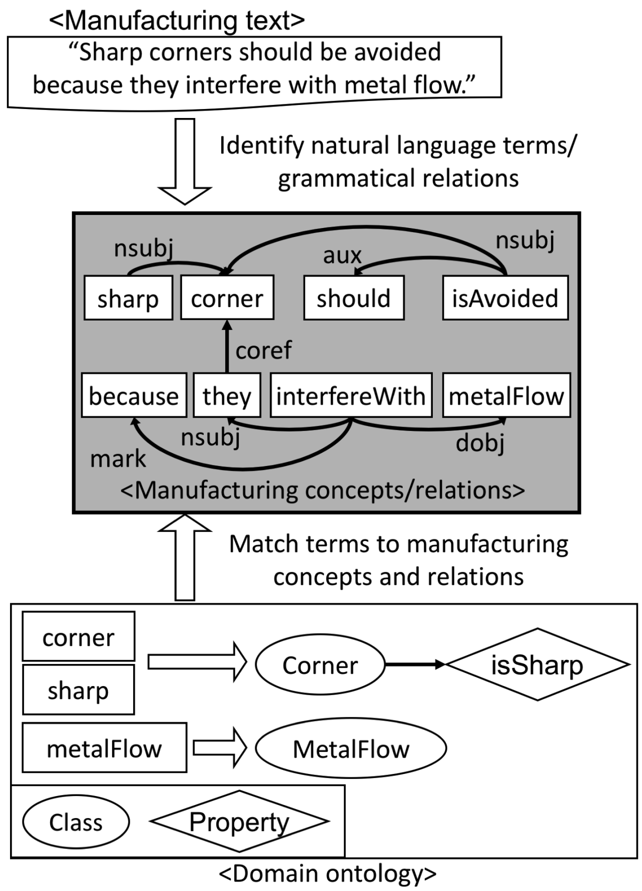



Figure 4.3: The manufacturing concepts/relations derived from the motivating example is shown. First, natural language terms/grammatical relations are identified, and then manufacturing-related terms are transformed to manufacturing concepts/relations defined in the manufacturing domain ontology.

### 4.6.1 Constraint Check Mechanism

In this dissertation, CBM is adopted to develop the constraints that identify the cases that need input text validation and provide relevant feedback. Figure 4.4 shows an example of the constraints and the overview of the constraint check mechanism. A constraint has two parts: satisfaction condition and feedback. Satisfaction condition part defines the pattern of manufacturing concepts/relations that indicate the necessity of input text validation. In the example of Figure 4.4, the satisfaction condition is the pattern that indicates the existence of a subordinate clause in the input text. Feedback part defines the advices that is given to the user when the constraint is triggered. Specifically, in this dissertation, the form of feedback is textual advice that provides the part of the input text needs validation, the detailed explanation of the issue, and the options to address the issue. In the example of Figure 4.4, the feedback suggests enclosing the subordinate clause as it contains potentially unnecessary information. Based on the feedback, the user can address the issue by modifying the input text to acquire correct formal manufacturing rules. Figure 4.5 shows the extraction of the correct formal manufacturing rule after modifying the input text. Specifically, it is shown that the unnecessary information, shown as the shaded region, is not included in the rule extracted from the modified sentence.

### 4.6.2 Constraint Category

Based on the examination of manufacturing texts and the extracted rules, a set of constraints are designed to identify the cases that need input text validation. The constraints are categorized into three categories based on

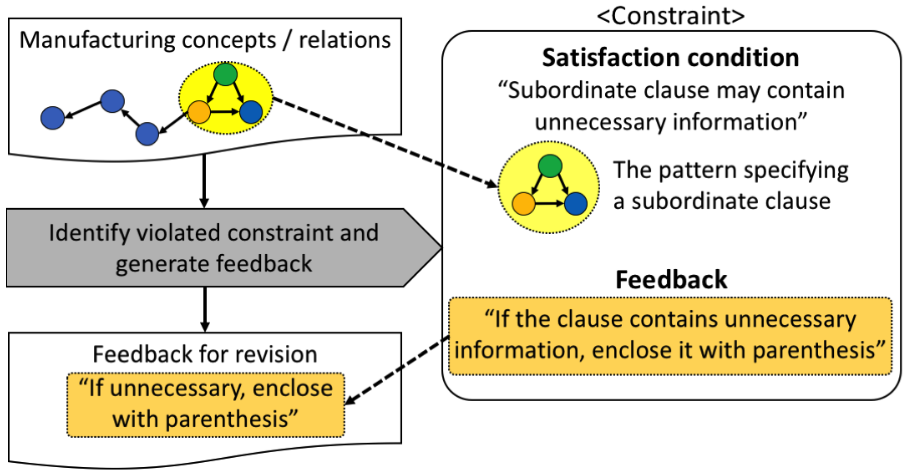



Figure 4.4: An example of constraint and the overview of constraint check mechanism are shown. The satisfaction condition of the constraint is utilized to identify if input text validation is needed, and textual feedback attached to the constraint is provided when the constraint is triggered

the types of issues: 1) the constraints identifying unnecessary information, 2) the constraints identifying semantic disagreement, and 3) the constraints identifying sentence without the explicit subject. It should be noted that the constraints and constraint categories are independent each other. In other words, a sentence may trigger multiple constraints in the same category or different categories.

**The constraints identifying unnecessary information** As shown in the motivating example (Figure 4.1), a manufacturing text may contain the information that is not necessary in the extracted rule, which makes the extracted rule redundant and not exactly correct. To address the issue, the first category of the constraints is designed to identify the existence of potentially unnecessary information. When the constraints are triggered, feedback suggests the user enclose the unnecessary part with parenthesis so that the part is explicitly

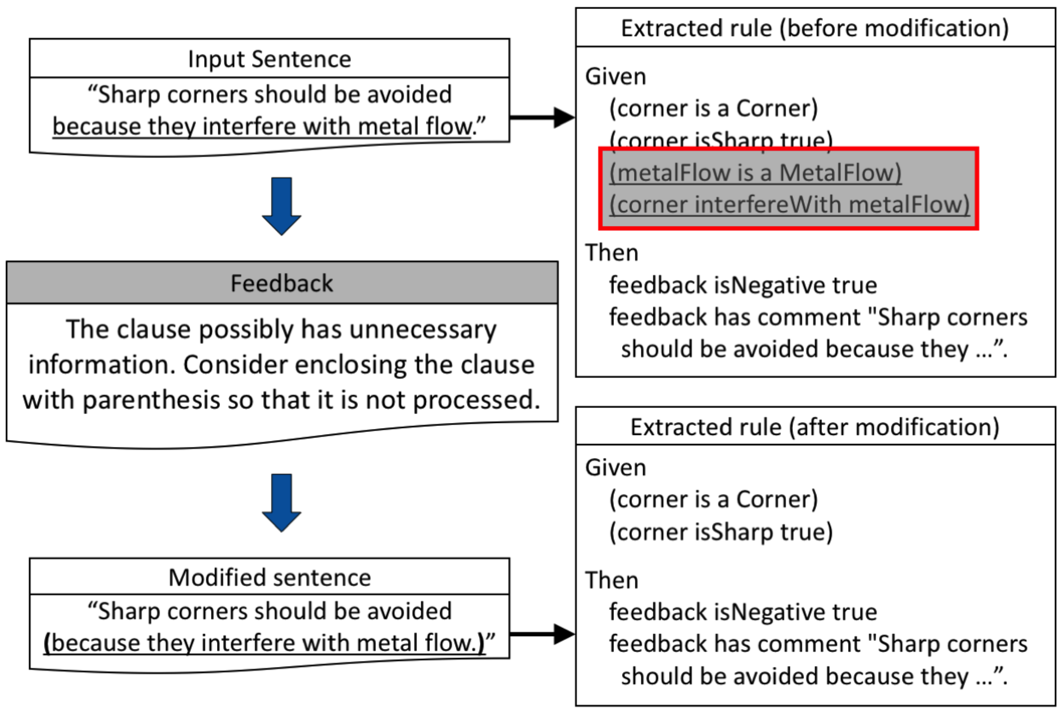
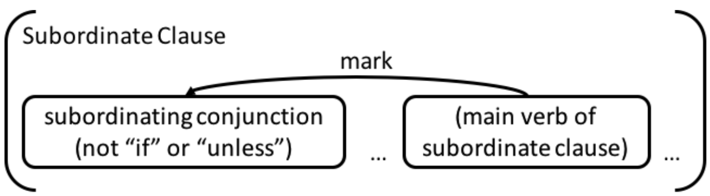
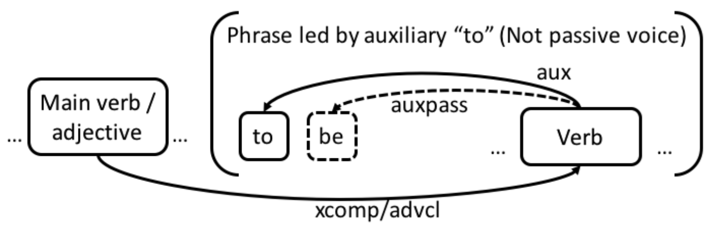
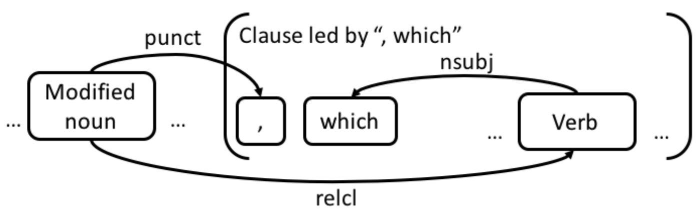



Figure 4.5: After modifying the input sentence based on the feedback, correct manufacturing rule can be extracted. Specifically, the unnecessary information, shown as the shaded region, is not included in the rule extracted from the modified sentence.

ignored in the rule extraction. The motivating example (Figure 4.1) belongs to this category, and the feedback suggested enclosing the subordinate clause "because they interfere metal flow" with parenthesis as it is unnecessary in the extracted rule. Table 4.1 shows the satisfaction conditions and the examples of the constraints in this category with the unnecessary information underlined. In the figures representing the patterns for the satisfaction conditions in Table 4.1, the solid line node and arrows are the manufacturing concepts/relations that should exist to trigger the constraint, and the dotted line nodes and arrows are the manufacturing concepts/relations that should not exist to trigger the constraint.

**The constraints identifying semantic disagreement** Sometimes, manufacturing text itself or the interpretation of manufacturing text represents semantically invalid relation between two concepts, which makes the extracted rule invalid. To address the issue, the second category of the constraints is designed to identify any semantic disagreement, such as mismatching between a feature and its attributes. When the constraints are triggered, the feedback suggests the user modify the sentence to address the semantic disagreement based on domain context in the manufacturing domain ontology. For example, the manufacturing text "If a blind hole is required, its length of depth should not exceed its diameter" describes that a "blind hole" have "length of depth", but it is semantically invalid as "length of depth" is redundant representation of "depth", which confuses rule extraction process. In this case, feedback is given to the user notifying that the attributes are not correctly related so that

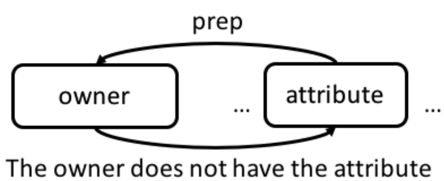
Table 4.1: The satisfaction conditions and examples of the first category of constraints, which identify the existence of unnecessary information, are shown with unnecessary information underlined.

| Satisfaction condition | Examples |
|---|---|
| Subordinate clause followed by a marker except for "if/unless"<br> | "Avoid knifelike edges of parts because they interfere with the smooth flow of material through the die." |
| Phrase involving to-infinitive (active voice)<br> | "To facilitate removal, the pattern must have taper or draft." |
| Non-definitive relative clause ", which"<br> | "Avoid sharp internal corners, which concentrate heat-treating stresses." |
| Other examples specifying unnecessary information | "Avoid blind areas, i.e., surfaces that are behind corners or protrusions." |

* The solid line nodes and arrows: Manufacturing concepts/relations that should exist to trigger the constraint

* The dotted line nodes and arrows: Manufacturing concepts/relations that should not exist to trigger the constraint.

Table 4.2: The satisfaction conditions and the examples of the second category of constraints, which identify the semantic disagreement between the concepts, are shown with relevant information underlined.

| Satisfaction condition | Examples |
|---|---|
| Semantic disagreement between an attribute and its owner   | "If a sharp internal corner is necessary, the included angle should be always more than 90." |

* The solid line nodes and arrows: Manufacturing concepts/relations that should exist to trigger the constraint

the user can modify the attribute based on the list of possible attributes in the manufacturing domain ontology. Table 4.2 shows the satisfaction conditions and the examples of the constraints in this category with the relevant part underlined. In the figures representing the patterns for the satisfaction conditions in Table 4.2, the solid line nodes and arrows are the manufacturing concepts/relations that should exist to trigger the constraint.

**The constraints identifying sentence without the explicit subject**

Since a formal manufacturing rule consists of the triples of "subject", "predicate", and "object/value", the formal rule extracted from the input sentence without an explicit subject may contain a *null* instance. To address the issue, the third category of the constraints is designed to identify the absence of an explicit subject. When the constraints are triggered, the feedback suggests the user check if the actual subject is correctly inferred, and re-perform the rule extraction after re-writing the sentence with the explicit subject. For example, in the manufacturing text "It is better not to have sharp corners on a formed part", the subject of the main verb is "it", which is a dummy pronoun. Thus, the subject of the main formal statement of the extracted rule is regarded as *null* as standard NLP techniques cannot infer the actual subject. For this example, the feedback suggests re-writing the sentence with the explicit subject, such as "part" or "corner". Table 4.3 shows the satisfaction conditions and the examples of the constraints in this category with the relevant part underlined. In the figures representing the patterns for the satisfaction conditions in Table 4.3, the solid line nodes and arrows are the manufacturing concepts/relations that should exist to trigger the constraint, and the dotted line nodes and arrows are the manufacturing concepts/relations that should not exist to trigger the constraint.

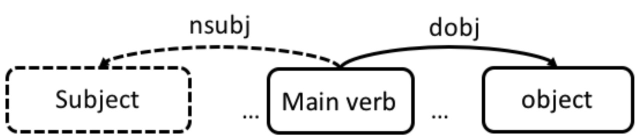
## 4.7 Implementation and Experiment

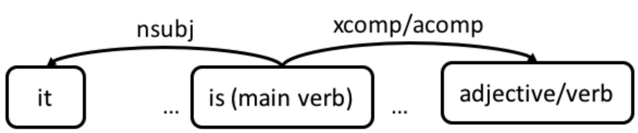
### 4.7.1 Experimental Setting

**Test scenario and dataset** As the test scenario, the formal manufacturing rule extraction framework is extended to adopt the feedback generation method. For the extension of the framework and experiment, the same dataset was used. For more details about the dataset creation and the list of sentences in the dataset, the readers are referred to Appendix A of this dissertation.

**Construction of manufacturing domain ontology** Similar to the ambiguity resolution framework, the manufacturing domain ontology used in the previous chapters do not require any modification to adopt feedback genera-

Table 4.3: The satisfaction conditions and the examples of the third category of constraints, which identify the absence of explicit subject of the main verb, are shown with relevant information underlined.

<table>
<tr><th>Satisfaction condition</th><th>Examples</th></tr>
<tr><td>Imperative mood


</td><td>“Avoid tapered-walled shells or flanged shells.”</td></tr>
<tr><td>The use pleonastic pronoun as subject


</td><td>“It is better not to have sharp corners on a formed part.”</td></tr>
</table>

* The solid line nodes and arrows: Manufacturing concepts/relations that should exist to trigger the constraint
* The dotted line nodes and arrows: Manufacturing concepts/relations that should not exist to trigger the constraint.

```
Corner is a type of ManufacturingFeature.
    isSharp describes Corner with a single value of type boolean.

MetalFlow is a type of MSDLPhysicalClass.
interfereWith describes Corner with values of type MetalFlow.
```

Figure 4.6: A part of manufacturing domain ontology used for this chapter is shown. The part specifies the manufacturing concepts/relations relevant to the motivating example of Figure 4.1.

tion. In other words, the feedback generation method can be applied to different types of semantics-based applications when the domain ontology provides enough semantics to perform feedback generation. Figure 4.6 shows the part of the manufacturing domain ontology corresponding to the domain context relevant to the motivating example (Figure 4.1).

### 4.7.2 Implementation

To verify the effectiveness of the feedback generation method, the previously developed Java application for formal manufacturing rule extraction is extended to adopt feedback generation. As mentioned in the previous chapter, the application utilizes NLP4J [79] and OpenNLP [80] as the NLP tools and utilizes Apache Jena Ontology API [83] for the ontology management and reasoning. The interface of the application is updated to provide the relevant feedback when the input text needs validation so that the user can address the issue. Figure 4.7 shows the screenshot of the application providing the relevant feedback for the motivating example. Specifically, the feedback suggests the user check if the subordinate clause contains unnecessary information and enclose the clause with parenthesis if the information is unnecessary.

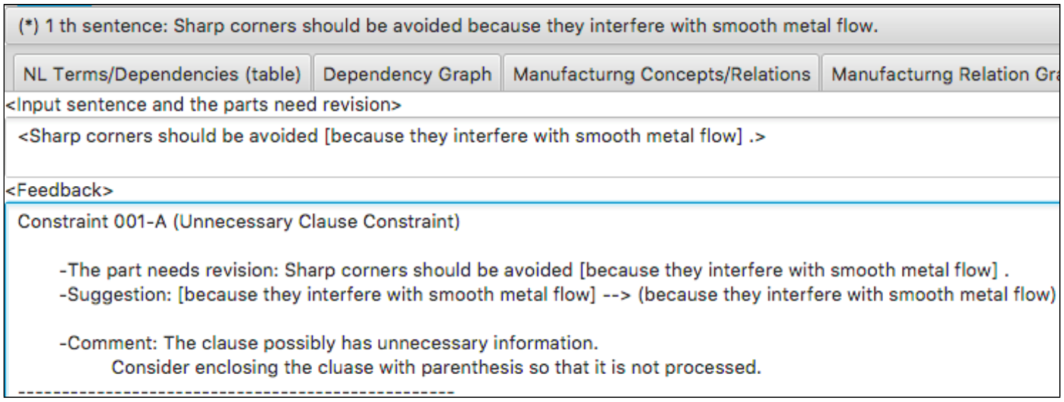
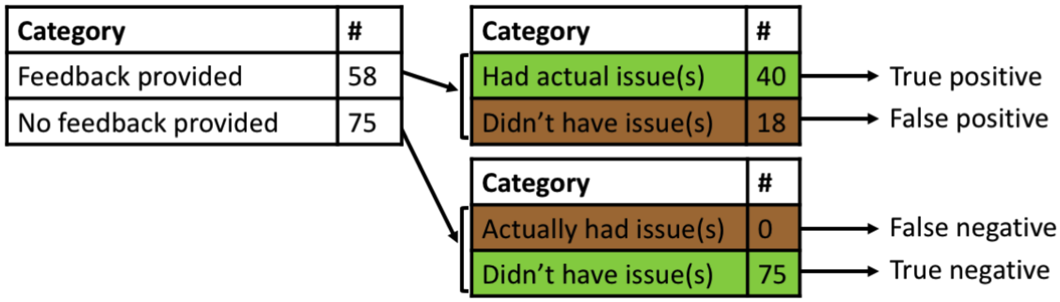



Figure 4.7: The screenshot of the updated Java application with feedback generation are shown. Specifically, the feedback provides the issue of the input text and the options for the user to address the issue.

### 4.7.3 Result and Discussion

The results of the feedback generation method are given in Figure 4.8. Figure 4.8 (a) shows the statistics of the results that indicate how many feedbacks are created and how many true/false positives and negatives there are. First, it is shown that relevant feedback has been provided to 58 cases, which include 40 relevant cases (true positive) and 18 irrelevant cases (false positive), as the constraints are designed in a conservative manner. Based on the feedback, correct rules are extracted from all the 40 cases, which need input text validation, after modifying input sentences based on the feedback. Second, there is no false negative, which is the case that feedback has not been provided while input text validation is needed. The results indicates that the feedback generation method can effectively help the user to validate the extracted rules by identifying most of the cases that need input text validation and providing useful feedback to modify the input text to acquire the desired rules. Figure 4.8 (b) shows the Precision, Recall, F-measure of the result, which can




(a)

| Precision | Recall | F-measure |
|---|---|---|
| 0.6897 | 1 | 0.8163 |

(b)

Figure 4.8: The results of feedback generation are shown as follows: (a) the number of the cases that feedback provided/not provided and the cases that need/don't need input text validation and (b) Precision, Recall, F-measure of the result.

be used as the baseline for the feedback generation for manufacturing rule extraction. In the rest of this section, a few examples of feedback generation for input text validation are shown. For the full list of the rule extraction results after applying feedback generation, the readers are referred to Appendix F of this dissertation.

**Feedback generation case 1** Figure 4.9 shows an example of feedback generation performed by the second category of constraints. In the example, the formal rule extracted from the input sentence "If a blind hole is required, its length of depth should not exceed its diameter", shown at the top-right, was not semantically valid due to the mismatching between the feature "blind hole" and the attribute "length of depth". Specifically, the extracted rule compares "length of blind hole" with "diameter of blind hole" and regards the owner of "depth" as *null*, which are not the intended behaviors. In this

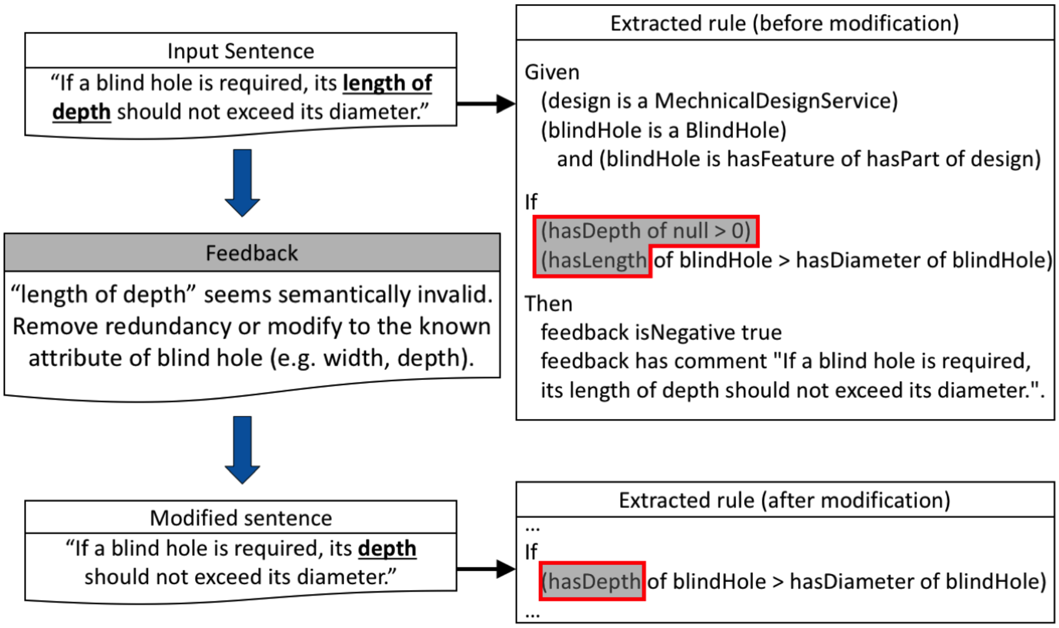



Figure 4.9: The example of feedback generation for input text validation is shown. The extracted rule was not semantically valid as "length of depth" is not a valid attribute of "blind hole". After modifying the sentence based on the feedback, the correct rule has been extracted as shown.

case, the feedback notifies that "length of depth" is not the valid attribute and suggests modifying the attribute based on the manufacturing domain ontology. Based on the feedback, the attribute is modified to "depth", and correct manufacturing rule has been extracted as shown at the bottom-right of Figure 4.9.

**Feedback generation case 2** Figure 4.10 shows another feedback generation performed by the third category of constraints. In the example, the rule extracted from the input sentence "It is better not to have sharp corners on a formed part", shown at the top-right, was not semantically valid due to the use of the dummy subject "it". Specifically, as the rule extraction framework

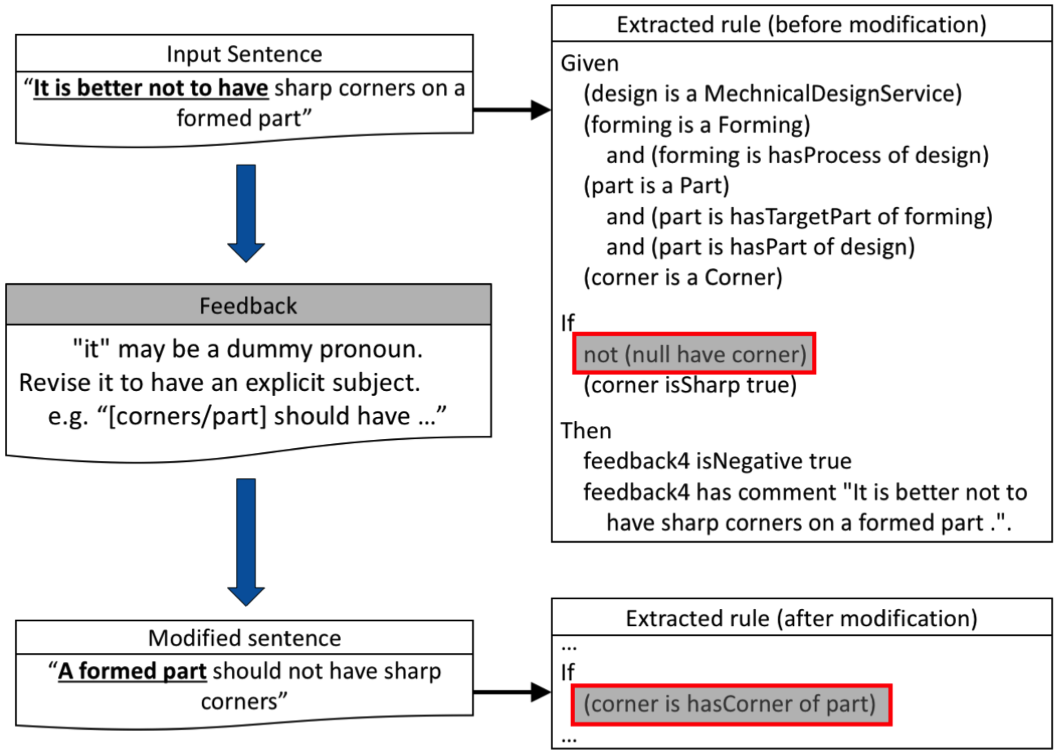



Figure 4.10: The example of feedback generation for input text validation is shown. The extract rule contained a *null* instance due to the use of a dummy subject "it". After modifying the sentence based on the feedback, the correct rule has been extracted as shown.

misses the owner of "corner", extracted rule regards the owner of "corner" as *null*, which is not the intended behavior. To address the issue, the feedback notifies that the sentence does not have an explicit subject and suggests modifying the sentence with an explicit subject such as "part" or "corners". Based on the feedback, the sentence is modified to "A formed part should not have sharp corners", and correct manufacturing rule has been extracted from the modified sentence as shown at the bottom-right of Figure 4.10.

## 4.8 Summary

In this chapter, the formal manufacturing rule extraction framework is extended to adopt the feedback generation method for input text validation. The feedback generation method is developed based on CBM-based, which relies on the constraints to identify the cases that need input text validation and provide relevant feedback. The effectiveness of the feedback generation method is demonstrated by generating relevant feedback for all the cases that need input text validation. Based on the feedback, the input texts are modified accordingly, and correct manufacturing rules are extracted from all the 40 cases not addressed by standard NLP techniques coupled with domain context. It should be also noted that the feedbacks are designed to provide the option that can be performed without learning formal structured languages. The results indicate that the feedback generation method can facilitate the validation of extracted rules by identifying the cases that need input text validation and helping the user to address the issue.

In the last chapter, the contributions of this dissertation, possible applications of the works of this dissertation, and future works to improve the results are presented.

# CHAPTER 5: CONCLUSION

In this chapter, the contributions of the works presented in this dissertation are summarized. Then the further applications and future challenges are presented.

## 5.1 Contributions

The overall goal of this dissertation is to overcome the formal knowledge acquisition gap that has limited the application of semantics-based approach in manufacturing field, especially semantics-based manufacturability analysis. To achieve the goal, this dissertation emphasizes the necessity of (semi-)automated formal manufacturing rule extraction mechanism. The following list specifies the major contributions of this dissertation. The detailed discussion of the contributions and the broader aspect of the contributions are followed.

- Design and development of the formal manufacturing rule extraction framework and the verification of its feasibility

- Extension of the rule extraction framework with the domain-specific ambiguity resolution method, which complements standard NLP techniques with domain context to address ambiguities in manufacturing text

- Extension of the rule extraction framework with the feedback generation method, which utilizes CBM to provide feedback for input text validation so that the user can modify the input text to acquire correct rules

Chapter 2 presents the research about extracting formal manufacturing rules from manufacturing text. Specifically, the semantic-based framework is developed to extract formal rules conforming to the manufacturing domain ontology. The feasibility of the framework is verified by acquiring formal manufacturing rules from unstructured text in a design for manufacturing handbook. To the best of knowledge, there is no prior research that focuses on the extraction of formal manufacturing rules from unstructured texts. The framework that been implemented leverages a blend of technologies that include standard NLP techniques coupled with semantics-based approaches that rely on the manufacturing domain ontology. In addition, the extension of the framework can be easily done by just updating the controlled vocabulary and manufacturing domain ontology. The extensibility of the framework is demonstrated by initially developing the framework using 3 sections of the handbook and extending into the rest of the handbook including 30 sections. The novelty of the developed framework also lies in the use of the controlled vocabulary and the formal manufacturing rule model for rule extraction. First, the controlled vocabulary is utilized to provide the explicit mapping between natural language texts and the manufacturing domain ontology. The explicit mapping helps the framework to align natural language terms into smaller sizes of formal classes or properties so that the extracted rule conforms to the manufacturing domain ontology. In addition, the formal manufacturing rule model is utilized to support the extraction of coherent manufacturing rules. Specifically, as the rule model defines the hierarchies and relations between the necessary components of a manufacturing rule for, the framework can derive the implicit concepts and relations based on the given information to make the extracted rule coherent.

Chapter 3 presents the research about extending the rule extraction framework to address domain-specific ambiguities that prevent the extraction of desired rules. Specifically, the domain-specific ambiguity resolution method is developed to complement standard NLP techniques with domain context. The effectiveness of the method is demonstrated by significantly improving rule extraction results from the same dataset. To the best of knowledge, there is no prior research that focuses on the ambiguity resolution of manufacturing text for rule extraction, and the existing ambiguity resolution approaches are not suitable for considering manufacturing domain context. The novelty of the method lies in the use of the manufacturing domain ontology as the mechanism providing domain context to address domain-specific ambiguity, and enable the extraction of desired manufacturing rules. In addition, as the ambiguity resolution method does not require a specially designed ontology for ambiguity resolution, the method can be easily incorporated into different semantics-based applications when domain ontology is available.

Chapter 4 presents the research about extending the rule extraction framework to address the issues of input text that prevent the extraction of desired rules. Specifically, the feedback generation method is developed to identify the cases that need input text validation and generate the relevant feedback to help the user address the input text issue. The effectiveness of the method is demonstrated from the same dataset by enabling the extraction of correct manufacturing rules from all the cases that need input text validation. To the best of knowledge, there is no prior research that focuses on the feedback generation for formal manufacturing rule extraction. The method is developed utilizing the existing feedback generation method, Constraint-Based Modeling, coupled with standard NLP techniques and the manufacturing domain ontology to generate the relevant feedback for formal manufacturing rule extraction. The results indicate that the feedback generation method is expected to facilitate the validation of the extracted rule by helping the user identify the cases that need input text validation and providing relevant feedback address the issue. Furthermore, the user needs not learn formal structured languages to follow the feedback as it suggests re-performing rule extraction after modifying the input text. Lastly, when domain ontology is available, the method can be easily extended by modeling the issues of input text using CBM.

Formal knowledge acquisition issue has been one of the major issues preventing the adoption of semantics-based approches in manufacturing field, especially semantics-based manufacturability analysis. To address the issue, this dissertation aims at closing the loop of formal manufacturing rule extraction by developing the rule extraction framework and extending the framework with the ambiguity resolution and feedback generation method. It is expected that the works of this dissertation can contribute to resolve the formal knowledge acquisition issue and thus address a critical need that has prevented a larger scale adoption of semantic technologies in manufacturing industry. For example, a wide range of textual manufacturing rules can be translated into formal manufacturing rules, and become accessible from CAD applications through the use of semantics-based manufacturability analysis [7].

## 5.2 Further Applications

Even though this dissertation demonstrates formal manufacturing rule extraction to facilitate the application of semantics-based manufacturability analysis, the extracted rules can be used for different applications. For example, formal manufacturing rules can be used as the foundation of semantics-based approaches in various stages of design and manufacturing process (e.g. design rule, product specification, safety requirement). These further applications of the formal manufacturing rules are available since the formal rules are written in the standard interoperable format and conform to MSDL, one of the high-level upper ontologies to describe various manufacturing services [2]. In addition, when domain ontology is available, the proposed procedure to develop the formal rule extraction framework can be extended to other areas, not restricted to manufacturing-related areas. The possible applications are summarized in the following list:

- The systematic procedures of utilizing standard NLP techniques coupled with semantics-based technology can be applied to different application domains when the translation of textual knowledge into semantically-usable formal knowledge is desired.

- The methodology of ambiguity resolution can also be applied to different application domains when the performance of standard NLP techniques is limited due to the lack of considering highly domain-specific context.

- This dissertation proposes that feedback generation can be used for input text validation, and thus facilitate the validation of the extracted rules. The procedure can be applied to different domains when developing fully automated validation mechanism is not feasible.

## 5.3 Future Works

This dissertation has made the contributions to the research area of the extraction of formal manufacturing rules from a manufacturing text. However, much work needs to be done for a practical scenario. The following lists are the identified challenges and future direction of the research:

**Facilitation of preprocessing** As mentioned in the preparation of the dataset, preprocessing is performed to convert the manufacturing text to XML format. Even though the preprocessing is manually done for this dissertation, the preprocessing becomes burdensome when the scale of the application becomes large. For a larger scale application, the preprocessing should be done (semi-)automatically to make the approach feasible.

**Improvement of algorithmic foundations** Several methods of the current framework can be further optimized for the accuracy of results. For example, while the semantic relations are estimated by checking the existence of the manufacturing concept or the relation between two concepts in the manufacturing domain ontology, more advanced techniques can be adapted to derive the likelihood of semantic relations between the concepts to improve the result provided by the framework.

**Handling broader (multiple sentences, paragraph, or entire text) contexts** While the works of this dissertation consider the context from the section titles that manufacturing text belongs to, the context across sentences, the paragraph, or the whole text is not considered. However, the actual manufacturing rule can span multiple sentences and the whole context may be important to deliver the intention of the author. Handling paragraphs to detect and identify the exact rule needs to be explored.

**Using external resource in rule extraction process** In this dissertation, external concept linking, suggested by Ameri et al. [61], is not performed in the construction of the controlled vocabulary. Connecting the framework with external publicly available large knowledge base is expected to improve the capability of the proposed framework.

**Developing NLP tools trained from manufacturing text** Since standard NLP techniques are used to analyze manufacturing text, the tools may provide wrong information which leads to the failure of formal manufacturing rule extraction. Developing NLP tools trained from manufacturing text can make the formal manufacturing rule extraction more robust.

**The effectiveness of the feedback** While the feedback generation method can generate relevant feedback for input text validation, the effectiveness of the feedback depends on the user. Specifically, the user is responsible for confirming the potential issue and modifying the input text based on the feedback. Therefore, it is difficult to estimate the effectiveness of the feedback since the same feedback may be utilized differently. In this context, there is a need for a mechanism to evaluate the effectiveness of the feedback to estimate the actual advantage provided by feedback generation.

## REFERENCES


[1] L. Ramos, "Semantic Web for manufacturing, trends and open issues Toward a state of the art," *Computers & Industrial Engineering*, vol. 90, pp. 444–460, Dec. 2015.

[2] F. Ameri and D. Dutta, "An Upper Ontology for Manufacturing Service Description," in *ASME 2006 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, vol. 3, Jan. 2006, pp. 651–661.

[3] S. A. Ghafour, P. Ghodous, B. Shariat, and E. Perna, "An ontology-based approach for "procedural cad models" data exchange.," *Frontiers in Artificial Intelligence and Applications*, vol. 143, pp. 251–259, 2006.

[4] S. Tessier and Y. Wang, "Ontology-based feature mapping and verification between CAD systems," *Advanced Engineering Informatics*, vol. 27, no. 1, pp. 76–92, Jan. 2013.

[5] J. Puttonen, A. Lobov, and M. Lastra, "Semantics-Based Composition of Factory Automation Processes Encapsulated by Web Services," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 4, pp. 2349–2359, Jan. 2013.

[6] T. Wang, S. Guo, and C.-G. Lee, "Manufacturing task semantic modeling and description in cloud manufacturing system," *The International Journal of Advanced Manufacturing Technology*, vol. 71, no. 9-12, pp. 2017–2031, Jan. 2014.

[7] A. Rangarajan, P. Radhakrishnan, A. Moitra, A. Crapo, and D. Robinson, "Manufacturability Analysis and Design Feedback System Devel-

oped Using Semantic Framework," in *ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Aug. 2013, pp. V004T05A001.

[8] L. Yu, S. Wang, and K. K. Lai, "A rough-set-refined text mining approach for crude oil market tendency forecasting," *International Journal of Knowledge and Systems Sciences*, vol. 2, pp. 33–46, 2005.

[9] N. Ur-Rahman and J. A. Harding, "Textual data mining for industrial knowledge management and text classification: A business oriented approach," *Expert Systems with Applications*, vol. 39, no. 5, pp. 4729–4739, 2012.

[10] D. L. McGuinness and F. v. Harmelen, *OWL Web Ontology Language Overview*. W3C Recommendation, 2004, Available at `https://www.w3.org/TR/owl-features/`.

[11] A. Crapo and A. Moitra, "Toward a unified english-like representation of semantic models, data, and graph patterns for subject matter experts," *International Journal of Semantic Computing*, vol. 07, no. 03, pp. 215–236, 2013.

[12] J. Bralla, *Design for Manufacturability Handbook*. McGraw Hill Professional, Aug. 1998.

[13] S. Conradson, L. Barford, W. Fisher, M. Weinstein, and J. Wilker, "Manufacturability tools: An engineer's use and needs," in *Electronic Manufacturing Technology Symposium, 1988, Design-to-Manufacturing Transfer Cycle. Fifth IEEE/CHMT International*, IEEE, 1988, pp. 155–158.

[14] *DFMPro - DFM and DFA software for NX, Creo and SolidWorks for automated Manufacturability checks*, Available at `http://dfmpro.geometricglobal.com/`.

[15] *3D CAD Design Software SolidWorks*, Available at `http://www.solidworks.com/`.

[16] *NX: Siemens PLM Software*, Available at `http://www.plm.automation.siemens.com/en_us/products/nx/`.

[17] *Boothroyd Dewhurst, Inc.* Available at `http://www.dfma.com/`.

[18] *Product Cost Management Software – aPriori*, Available at `http://www.apriori.com/`.

[19] K. Scherer, P Buchberger, H Eggert, P Stiller, and U Stucky, "Knowledge based support for manufacturing of microstructures," *Microsystem technologies*, vol. 2, no. 4, pp. 167–170, 1996.

[20] B Arezoo, K Ridgway, and A. Al-Ahmari, "Selection of cutting tools and conditions of machining operations using an expert system," *Computers in Industry*, vol. 42, no. 1, pp. 43–58, 2000.

[21] D. Brissaud and S. Tichkiewitch, "Innovation and manufacturability analysis in an integrated design context," *Computers in Industry*, vol. 43, no. 2, pp. 111–121, 2000.

[22] D. S. Chan, "Expert system for product manufacturability and cost evaluation," *Materials and Manufacturing Processes*, vol. 18, no. 2, pp. 313–322, 2003.

[23] D. V. Jacob, K. Ramana, and P. Rao, "Automated manufacturability assessment of rotational parts by grinding," *International journal of production research*, vol. 42, no. 3, pp. 505–519, 2004.

[24] S. Kumar, R. Singh, and G. Sekhon, "Cckbs: A component check knowledge-based system for assessing manufacturability of sheet metal parts," *Journal of Materials Processing Technology*, vol. 172, no. 1, pp. 64–69, 2006.

[25] K. Ramana, M. Singh, A. Gupta, R. Dey, A. Kapoor, and P. Rao, "A manufacturability advisor for spun and rollformed sheet metal components," *The International Journal of Advanced Manufacturing Technology*, vol. 28, no. 3, pp. 249–254, 2006.

[26] T Giannakakis and G. Vosniakos, "Sheet metal cutting and piercing operations planning and tools configuration by an expert system," *The International Journal of Advanced Manufacturing Technology*, vol. 36, no. 7, pp. 658–670, 2008.

[27] Y.-M. Chen, R. A. Miller, and K. Sevenler, "Knowledge-based manufacturability assessment: An object-oriented approach," *Journal of intelligent Manufacturing*, vol. 6, no. 5, pp. 321–337, 1995.

[28] M. S. Amalnik and J. McGeough, "Intelligent concurrent manufacturability evaluation of design for electrochemical machining," *Journal of materials processing technology*, vol. 61, no. 1-2, pp. 130–139, 1996.

[29] S. K. Gupta, D. S. Nau, and W. C. Regli, "Imacs: A case study in real-world planning," *IEEE Intelligent Systems and their Applications*, vol. 13, no. 3, pp. 49–60, 1998.

[30] R. E. Giachetti and M. I. Alvi, "An object-oriented information model for manufacturability analysis of printed circuit board fabrication," *Computers in industry*, vol. 45, no. 2, pp. 177–196, 2001.

[31] A. Venkatachalam, J. M. Mellichamp, and D. M. Miller, "A knowledge-based approach to design for manufacturability," *Journal of Intelligent Manufacturing*, vol. 4, no. 5, pp. 355–366, 1993.

[32] H. Jia, S. Ong, J. Fuh, Y. Zhang, and A. Nee, "An adaptive and upgradable agent-based system for coordinated product development and manufacture," *Robotics and Computer-Integrated Manufacturing*, vol. 20, no. 2, pp. 79–90, 2004.

[33] K. Ramana and P. Rao, "Automated manufacturability evaluation system for sheet metal components in mass production," *International Journal of Production Research*, vol. 43, no. 18, pp. 3889–3913, 2005.

[34] S. Ong and L. Chew, "Evaluating the manufacturability of machined parts and their setup plans," *International Journal of Production Research*, vol. 38, no. 11, pp. 2397–2415, 2000.

[35] S. Ong, M. Sun, and A. Nee, "A fuzzy set ahp-based dfm tool for rotational parts," *Journal of Materials Processing Technology*, vol. 138, no. 1, pp. 223–230, 2003.

[36] M Korosec, J Balic, and J Kopac, "Neural network based manufacturability evaluation of free form machining," *International Journal of Machine Tools and Manufacture*, vol. 45, no. 1, pp. 13–20, 2005.

[37] Z. Zhao and J. J. Shah, "Domain independent shell for dfm and its application to sheet metal forming and injection molding," *Computer-Aided Design*, vol. 37, no. 9, pp. 881–898, 2005.

[38] M. C. Yang, W. H. Wood, and M. R. Cutkosky, "Design information retrieval: A thesauri-based approach for reuse of informal design information," *Engineering with computers*, vol. 21, no. 2, pp. 177–192, 2005.

[39] A. Riel and P. Boonyasopon, "A Knowledge Mining Approach to Document Classification," *The Asian International Journal of Science and Technology in Production and Manufacturing*, vol. 2, no. 3, pp. 1–10, Jul. 2009.

[40] "Indutech (Pty) Ltd," [Online]. Available: `{http://www.indutech.co.za/}`.

[41] P Boonyasopon, A Riel, W Uys, L Louw, S Tichkiewitch, and N du Preez, "Automatic knowledge extraction from manufacturing research publications," *Cirp Annals-Manufacturing Technology*, vol. 60, no. 1, pp. 477–480, 2011.

[42] R. Mihalcea and A. Csomai, "Wikify!: linking documents to encyclopedic knowledge," in *CIKM '07 Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, Nov. 2007, pp. 233–242.

[43] P. Y. Shotorbani, F. Ameri, B. Kulvatunyou, and N. Ivezic, "A hybrid method for manufacturing text mining based on document clustering and topic modeling techniques," in *IFIP International Conference on*

*Advances in Production Management Systems*, Springer, 2016, pp. 777–786.

[44] P. Yazdizadeh and F. Ameri, "A Text Mining Technique for Manufacturing Supplier Classification," in *ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Aug. 2015, V01BT02A036.

[45] Z. Kaijun and T. Yifei, "Research on manufacturing text classification based on improved genetic algorithm," *Brazilian Archives of Biology and Technology*, vol. 59, no. SPE, 2016.

[46] Z. Li, M. Liu, D. C. Anderson, and K. Ramani, "Semantics-Based Design Knowledge Annotation and Retrieval," in *ASME 2005 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Jan. 2005, pp. 799–808.

[47] M. C. Yang, W. H. Wood III, and M. R. Cutkosky, "Design information retrieval: a thesauri-based approach for reuse of informal design information," *Engineering with computers*, vol. 21, no. 2, pp. 177–192, Nov. 2005.

[48] Z. Li and K. Ramani, "Ontology-based design information extraction and retrieval," *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, vol. 21, no. 02, pp. 137–154, Mar. 2007.

[49] S. Choi, H. Park, D. Kang, J. Y. Lee, and K. Kim, "An sao-based text mining approach to building a technology tree for technology planning," *Expert Systems with Applications*, vol. 39, no. 13, pp. 11 443–11 455, 2012.

[50] H. Cheong, W. Li, A. Cheung, A. Nogueira, and F. Iorio, "Automatic Extraction of Function Knowledge From Text," in *ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 2015, pp. V02AT03A013 month = aug.

[51] H. Cheong, W. Li, and F. Iorio, "Automated Extraction of System Structure Knowledge From Text," in *ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Aug. 2016, pp. V02AT03A011.

[52] J. Hirtz, R. B. Stone, D. A. McAdams, S. Szykman, and K. L. Wood, "A functional basis for engineering design: Reconciling and evolving previous efforts," *Research in Engineering Design*, vol. 13, no. 2, pp. 65–82, Mar. 2002.

[53] G. A. Miller, "WordNet: a lexical database for English," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, Nov. 1995.

[54] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," *ArXiv.org*, Jan. 2013. arXiv: `1301.3781v3 [cs.CL]`.

[55] L. Lan, Y. Liu, and W. F. Lu, "Discovering a Hierarchical Design Process Model Using Text Mining," in *ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Aug. 2016, pp. V01BT02A024.

[56] G. Wang, X. Tian, J. Geng, R. Evans, and S. Che, "Extraction of Principle Knowledge from Process Patents for Manufacturing Process Innovation," *Procedia CIRP*, vol. 56, pp. 193–198, 2016.

[57] S. M. Jeon, J. H. Lee, G. J. Hahm, and H. W. Suh, "Automatic CAD model retrieval based on design documents using semantic processing and rule processing," *Computers in Industry*, vol. 77, pp. 29–47, Apr. 2016.

[58] S. Ahmed, S. Kim, and K. M. Wallace, "A Methodology for Creating Ontologies for Engineering Design," *Journal of Computing and Information Science in Engineering*, vol. 7, no. 2, pp. 132–140, 2007.

[59] Z. Li, V. Raskin, and K. Ramani, "A Methodology of Engineering Ontology Development for Information Retrieval," in *ICED07 16th International Conference of Engineering Design*, Aug. 2007.

[60] Z. Li, M. C. Yang, and K. Ramani, "A methodology for engineering ontology acquisition and validation," *AI EDAM*, vol. 23, no. 1, pp. 37–51, 2009.

[61] F. Ameri, B. Kulvatunyou, N. Ivezic, and K. Kaikhah, "Ontological conceptualization based on the skos," *Journal of Computing and Information Science in Engineering*, vol. 14, no. 3, p. 031 006, Sep. 2014.

[62] J. Kang and J. K. Lee, "Rule identification from Web pages by the XRML approach," *Decision Support Systems*, vol. 41, no. 1, pp. 205–227, Nov. 2005.

[63] J. K. Lee and M. M. Sohn, "The eXtensible Rule Markup Language," *Communications of the ACM*, vol. 46, no. 5, pp. 59–64, May 2003.

[64] S. Park and J. K. Lee, "Rule identification using ontology while acquiring rules from Web pages," *International Journal of Human-Computer Studies*, vol. 65, no. 7, pp. 659–673, Jul. 2007.

[65] A. Singh and B. Muruganantham, "Automatic Rule Extraction from Frequent Rules In Similar Network Sites," *International Journal of Computer Trends and Technology*, vol. 4, pp. 292–295, 3 2013.

[66] D. Santhakumar and S. Rajasekar, "Automatic rule retrieval from websites using ontology and text mining," *International Journal of Emerging Technology in Computer Science Electronics (IJETCSE)*, vol. 13, no. 2, pp. 29–34, 2015.

[67] S. Hassanpour, M. J. O'Connor, and A. K. Das, "A Framework for the Automatic Extraction of Rules from Online Text," in *Proceedings of the International Symposium on Rule Interchange and Application: 2011*, 2011, pp. 266–280.

[68] X. Ye, "Identify the semantic meaning of service rules with natural language processing," in *Parallel and Distributed Computing, Applications and Technologies (PDCAT), 2016 17th International Conference on*, IEEE, 2016, pp. 63–68.

[69] X. Ye and P. Zhao, "Converting service rules to semantic rules," in *Services Computing (SCC), 2016 IEEE International Conference on*, IEEE, 2016, pp. 752–759.

[70] A.-P. Manine, É. Alphonse, and P. Bessières, "Learning ontological rules to extract multiple relations of genic interactions from text," *International Journal of Medical Informatics*, vol. 78, pp. 31–38, 2009.

[71] J. Hakenberg, C. Plake, U. Leser, H. Kirsch, and D. Rebholz-Schuhmann, "Lll05 challenge: Genic interaction extraction-identification of language patterns based on alignment and finite state automata," in *Proceedings of the 4th Learning Language in Logic workshop (LLL05)*, Citeseer, 2005, pp. 38–45.

[72] W.-J. Hou and H.-Y. Chen, "Rule extraction in gene-disease relationship discovery," *Gene*, vol. 518, no. 1, pp. 132–138, 2013.

[73] A. Hamosh, A. F. Scott, J. Amberger, C. Bocchini, D. Valle, and V. A. McKusick, "Online mendelian inheritance in man (OMIM), a knowledgebase of human genes and genetic disorders," *Nucleic Acids Research*, vol. 30, no. 1, pp. D514–D517, Jan. 2002.

[74] S. Muggleton and L. de Raedt, "Inductive logic programming: Theory and methods," *The Journal of Logic Programming*, vol. 19, pp. 629–679, May 1994.

[75] A. Boufrida and Z. Boufaida, "Automatic rules extraction from medical texts," in *IWAISE 2014 International Workshop on Advanced Information Systems for Enterprises*, 2014, pp. 29–33.

[76] T. Bray, J. Paoli, C. M. Sperberg-McQueen, E. Maler, F. Yergeau, and J. Cowan, *Extensible Markup Language (XML) 1.1 (Second Edition)*. W3C—World Wide Web Consortium, 2006, Available at `https://www.w3.org/TR/xml11/`.

[77] A. Miles and S. Bechhofer, *SKOS Simple Knowledge Organization System Reference*. W3C—World Wide Web Consortium, 2009, Available at `https://www.w3.org/TR/skos-reference/`.

[78] S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python*, ser. Analyzing Text with the Natural Language Toolkit. O'Reilly Media, Inc., Jun. 2009.

[79] *The Natural Language Processing for JVM languages (NLP4J)*, Available at `https://emorynlp.github.io/nlp4j/`.

[80] *Apache OpenNLP*, Available at `https://opennlp.apache.org/`.

[81] F. Manola, E. Miller, and B. McBride, *RDF Primer*. W3C—World Wide Web Consortium, 2004, Available at `https://www.w3.org/TR/2004/REC-rdf-primer-20040210/`.

[82] D. Brickley, R. Guha, and B. McBride, *RDF Vocabulary Description Language 1.0: RDF Schema*. W3C—World Wide Web Consortium, 2004, Available at `https://www.w3.org/TR/2004/REC-rdf-schema-20040210/`.

[83] *Apache Jena*, Available at `https://jena.apache.org/`.

[84] A. Franz, *Automatic ambiguity resolution in natural language processing: An empirical approach*. Springer Science & Business Media, 1996, vol. 1171.

[85] R. Mitkov, "Anaphora resolution: The state of the art," Tech. Rep., 1999.

[86] V. Ng, "Supervised noun phrase coreference research: The first fifteen years," in *Proceedings of the 48th annual meeting of the association for computational linguistics*, Association for Computational Linguistics, 2010, pp. 1396–1411.

[87] M. Poesio, R. Stuckardt, and Y. Versley, *Anaphora resolution: Algorithms, resources, and applications*. Springer, 2016.

[88] S. Lappin and H. J. Leass, "An algorithm for pronominal anaphora resolution," *Computational linguistics*, vol. 20, no. 4, pp. 535–561, 1994.

[89] R. Mitkov, "An integrated model for anaphora resolution," in *Proceedings of the 15th conference on Computational linguistics-Volume 2*, Association for Computational Linguistics, 1994, pp. 1170–1176.

[90] ——, "Robust pronoun resolution with limited knowledge," in *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 2*, Association for Computational Linguistics, 1998, pp. 869–875.

[91] J. Hobbs, "Resolving pronoun references," *Lingua 44*, pp. 311–338, 1978.

[92] H. Alshawi, *Memory and context for language interpretation*. Cambridge: Cambridge University Press, 1987.

[93] B. J. Grosz, A. K. Joshi, and S Weinstein, *Towards a computational theory of discourse interpretation, Ms.* 1986.

[94] S. E. Brennan, M. W. Friedman, and C. J. Pollard, "A centering approach to pronouns," in *Proceedings of the 25th annual meeting on Association for Computational Linguistics*, Association for Computational Linguistics, 1987, pp. 155–162.

[95] M. Strube, "Never look back: An alternative to centering," in *Proceedings of the 17th international conference on Computational linguistics-*

*Volume 2*, Association for Computational Linguistics, 1998, pp. 1251–1257.

[96] J. R. Tetreault, "A corpus-based evaluation of centering and pronoun resolution," *Computational Linguistics*, vol. 27, no. 4, pp. 507–520, 2001.

[97] ——, "Analysis of syntax-based pronoun resolution methods," in *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, Association for Computational Linguistics, 1999, pp. 602–605.

[98] I. Dagan and A. Itai, "Automatic processing of large corpora for the resolution of anaphora references," in *Proceedings of the 13th conference on Computational linguistics-Volume 3*, Association for Computational Linguistics, 1990, pp. 330–332.

[99] M. Paul, K. Yamamoto, and E. Sumita, "Corpus-based anaphora resolution towards antecedent preference," in *Proceedings of the Workshop on Coreference and its Applications*, Association for Computational Linguistics, 1999, pp. 47–52.

[100] V. Ng and C. Cardie, "Improving machine learning approaches to coreference resolution," in *Proceedings of the 40th annual meeting on association for computational linguistics*, Association for Computational Linguistics, 2002, pp. 104–111.

[101] J. R. Quinlan, *C4. 5: Programs for machine learning*. Elsevier, 2014.

[102] R. Iida, K. Inui, and Y. Matsumoto, "Anaphora resolution by antecedent identification followed by anaphoricity determination," *ACM*

*Transactions on Asian Language Information Processing (TALIP)*, vol. 4, no. 4, pp. 417–434, 2005.

[103] H. Yang, A. de Roeck, V. Gervasi, A. Willis, and B. Nuseibeh, "Extending Nocuous Ambiguity Analysis for Anaphora in Natural Language Requirements," in *Proceedings of the 18th International Requirements Engineering Conference*, IEEE, 2010, pp. 25–34.

[104] ——, "Analysing anaphoric ambiguity in natural language requirements," *Requirements engineering*, vol. 16, no. 3, pp. 163–189, 2011.

[105] S. Broscheit, M. Poesio, S. P. Ponzetto, K. J. Rodriguez, L. Romano, O. Uryupina, Y. Versley, and R. Zanoli, "Bart: A multilingual anaphora resolution system," in *Proceedings of the 5th international workshop on semantic evaluation*, Association for Computational Linguistics, 2010, pp. 104–107.

[106] O. Uryupina, "Corry: A system for coreference resolution," in *Proceedings of the 5th international workshop on semantic evaluation*, Association for Computational Linguistics, 2010, pp. 100–103.

[107] M. Recasens, L. Màrquez, E. Sapena, M. A. Martí, M. Taulé, V. Hoste, M. Poesio, and Y. Versley, "Semeval-2010 task 1: Coreference resolution in multiple languages," in *Proceedings of the 5th International Workshop on Semantic Evaluation*, Association for Computational Linguistics, 2010, pp. 1–8.

[108] S. Pradhan, L. Ramshaw, M. Marcus, M. Palmer, R. Weischedel, and N. Xue, "Conll-2011 shared task: Modeling unrestricted coreference in ontonotes," in *Proceedings of the Fifteenth Conference on Computa-*

*tional Natural Language Learning: Shared Task*, Association for Computational Linguistics, 2011, pp. 1–27.

[109] K.-W. Chang, R. Samdani, A. Rozovskaya, N. Rizzolo, M. Sammons, and D. Roth, “Inference protocols for coreference resolution,” in *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task*, Association for Computational Linguistics, 2011, pp. 40–44.

[110] A. Björkelund and P. Nugues, “Exploring lexicalized features for coreference resolution,” in *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task*, Association for Computational Linguistics, 2011, pp. 45–50.

[111] C. N. dos Santos and D. L. Carvalho, “Rule and tree ensembles for unrestricted coreference resolution,” in *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task*, Association for Computational Linguistics, 2011, pp. 51–55.

[112] A. Zelaia, O. Arregi, and B. Sierra, “A multi-classifier approach to support coreference resolution in a vector space model.,” in *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing, VS@NAACL-HLT 2015, June 5, 2015, Denver, Colorado, USA*, 2015, pp. 17–24.

[113] H. Lee, A. Chang, Y. Peirsman, N. Chambers, M. Surdeanu, and D. Jurafsky, “Deterministic coreference resolution based on entity-centric, precision-ranked rules,” *Computational Linguistics*, vol. 39, no. 4, pp. 885–916, 2013.

[114] S. P. Ponzetto and M. Strube, "Exploiting semantic role labeling, wordnet and wikipedia for coreference resolution," in *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, Association for Computational Linguistics, 2006, pp. 192–199.

[115] S. S. Pradhan, W. H. Ward, K. Hacioglu, J. H. Martin, and D. Jurafsky, "Shallow semantic parsing using support vector machines," in *HLT-NAACL*, 2004, pp. 233–240.

[116] V. Bryl, C. Giuliano, L. Serafini, and K. Tymoshenko, "Supporting natural language processing with background knowledge: Coreference resolution case," *The Semantic Web–ISWC 2010*, pp. 80–95, 2010.

[117] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, and Z. Ives, "Dbpedia: A nucleus for a web of open data," in *In 6th Intl Semantic Web Conference, Busan, Korea*, Springer, 2007, pp. 11–15.

[118] F. M. Suchanek, G. Kasneci, and G. Weikum, "Yago: A core of semantic knowledge," in *Proceedings of the 16th international conference on World Wide Web*, ACM, 2007, pp. 697–706.

[119] C. Giuliano, A. M. Gliozzo, and C. Strapparava, "Kernel methods for minimally supervised wsd," *Computational Linguistics*, vol. 35, no. 4, pp. 513–528, 2009.

[120] O. Uryupina, M. Poesio, C. Giuliano, and K. Tymoshenko, "Disambiguation and filtering methods in using web knowledge for coreference resolution.," in *FLAIRS Conference*, vol. 68, 2011, pp. 317–322.

[121] A. Rahman and V. Ng, "Coreference resolution with world knowledge," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, Association for Computational Linguistics, 2011, pp. 814–824.

[122] C. F. Baker, C. J. Fillmore, and J. B. Lowe, "The berkeley framenet project," in *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 1*, Association for Computational Linguistics, 1998, pp. 86–90.

[123] S. Kurohashi and M. Nagao, "Dynamic programming method for analyzing conjunctive structures in japanese," in *Proceedings of the 14th conference on Computational linguistics-Volume 1*, Association for Computational Linguistics, 1992, pp. 170–176.

[124] P Resnik, "Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language," *Journal of Artificial Intelligence Research*, vol. 11, pp. 95–130, 1999.

[125] M. Goldberg, "An unsupervised model for statistically determining coordinate phrase attachment," in *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, Association for Computational Linguistics, 1999, pp. 610–614.

[126] P. Nakov and M. Hearst, "Using the web as an implicit training set: Application to structural ambiguity resolution," in *Proceedings of the*

*conference on Human Language Technology and Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, 2005, pp. 835–842.

[127] F. Chantree, A. Kilgarriff, A. Roeck, and A. Willis, "Disambiguating coordinations using word distribution information," in *In Proceedings of RANLP2005*, pp. 21–23.

[128] P. V. Ogren, "Improving syntactic coordination resolution using language modeling," in *Proceedings of the NAACL HLT 2010 Student Research Workshop*, Association for Computational Linguistics, 2010, pp. 1–6.

[129] A. Hanamoto, T. Matsuzaki, and J. Tsujii, "Coordination structure analysis using dual decomposition," in *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, Association for Computational Linguistics, 2012, pp. 430–438.

[130] M. Collins, "Head-driven statistical models for natural language parsing," *Computational linguistics*, vol. 29, no. 4, pp. 589–637, 2003.

[131] R. Agarwal and L. Boggess, "A simple but useful approach to conjunct identification," in *Proceedings of the 30th annual meeting on Association for Computational Linguistics*, Association for Computational Linguistics, 1992, pp. 15–21.

[132] A. Okumura and K. Muraki, "Symmetric pattern matching analysis for english coordinate structures," in *Proceedings of the fourth conference*

*on Applied natural language processing*, Association for Computational Linguistics, 1994, pp. 41–46.

[133] V. Rus, D. I. Moldovan, and O. Bolohan, "Bracketing compound nouns for logic form derivation," in *FLAIRS Conference*, 2002, pp. 198–202.

[134] J. Nilsson, J. Nivre, and J. Hall, "Graph transformations in data-driven dependency parsing," in *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, 2006, pp. 257–264.

[135] J Hajic, "Building a syntactically annotated corpus: The prague dependency treebank, 1998," in. Karolinum, 1998, pp. 106–132.

[136] D. Hogan, "Coordinate noun phrase disambiguation in a generative parsing model," in *ACL 2007, Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, 2007.

[137] K. Hara, M. Shimbo, H. Okuma, and Y. Matsumoto, "Coordinate structure analysis with global structural constraints and alignment-based local features," in *Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP*, Association for Computational Linguistics, vol. 2, 2009, pp. 967–975.

[138] H. Yang, A. Willis, A. De Roeck, and B. Nuseibeh, "Automatic detection of nocuous coordination ambiguities in natural language requirements," in *Proceedings of the IEEE/ACM international conference on Automated software engineering*, ACM, 2010, pp. 53–62.

[139] H. Keuning, J. Jeuring, and B. Heeren, "Towards a systematic review of automated feedback generation for programming exercises," in *Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education*, ACM, 2016, pp. 41–46.

[140] A. T. Corbett, J. R. Anderson, and E. G. Patterson, "Student modeling and tutoring flexibility in the lisp intelligent tutoring system," *Intelligent tutoring systems: At the crossroads of artificial intelligence and education*, pp. 83–106, 1990.

[141] S. Ohlsson, "Constraint-based student modeling," in *Student modelling: The key to individualized knowledge-based instruction*. Springer, 1994, pp. 167–189.

[142] A. Mitrovic, P. Suraweera, B. Martin, and A. Weerasinghe, "Db-suite: Experiences with three intelligent, web-based database tutors," *Journal of Interactive Learning Research*, vol. 15, no. 4, pp. 409–432, 2004.

[143] N.-T. Le and N. Pinkwart, "Incom: A web-based homework coaching system for logic programming," in *Conference on Cognition and Exploratory Learning in Digital Age*, 2011, pp. 43–50.

[144] R. Matloobi, M. Blumenstein, and S. Green, "Extensions to generic automated marking environment: Game-2," in *Interactive Computer Aided Learning Conference*, 2009, pp. 1069–1076.

[145] E. Odekirk-Hash and J. L. Zachary, "Automated feedback on programs means students need less help from teachers," in *ACM SIGCSE Bulletin*, ACM, vol. 33, 2001, pp. 55–59.

[146] S. Xu and Y. San Chee, "Transformation-based diagnosis of student programs for programming tutoring systems," *IEEE Transactions on Software Engineering*, vol. 29, no. 4, pp. 360–384, 2003.

[147] H. Ueno, "A generalized knowledge-based approach to comprehend pascal and c programs," *IEICE Transactions on Information and Systems*, vol. 83, no. 4, pp. 591–598, 2000.

[148] W. L. Johnson and L. A. Gladwin, "Intention-based diagnosis of novice programming errors," *IEEE Expert*, vol. 2, no. 3, pp. 94–94, 1987.

[149] H. C. Lane and K. VanLehn, "Teaching the tacit knowledge of programming to noviceswith natural language tutoring," *Computer Science Education*, vol. 15, no. 3, pp. 183–201, 2005.

[150] C. MacNish, "Machine learning and visualisation techniques for inferring logical errors in student code submissions," in *ICALT-2002: Proc. 2nd IEEE Int. Conf. on Advanced Learning Technologies*, 2002, pp. 317–321.

[151] A. Ferreira and J. Atkinson, "Designing a feedback component of an intelligent tutoring system for foreign language," *Knowledge-Based Systems*, vol. 22, no. 7, pp. 496–501, 2009.

[152] F. Gutierrez and J. Atkinson, "Adaptive feedback selection for intelligent tutoring systems," *Expert Systems with Applications*, vol. 38, no. 5, pp. 6146–6152, 2011.

[153] N Noriko, "Banzai: Computer assisted sentence production practice with intelligent feedback," *Computer assisted system for teaching and learning Japanese, 2002*, 2002.

[154] M. Dzikovska, N. Steinhauser, E. Farrow, J. Moore, and G. Campbell, “Beetle ii: Deep natural language understanding and automatic feedback generation for intelligent tutoring in basic electricity and electronics,” *International Journal of Artificial Intelligence in Education*, vol. 24, no. 3, pp. 284–332, 2014.

# APPENDIX A: THE DETAILS OF DATASET CREATION AND FULL SENTENCE LIST

In this dissertation, the dataset consists of the sentences with rule-like information, chosen from the subsection "Design Recommendation" of each section of the design for manufacturing handbook written by Bralla [12]. Please note that the goal of this research is not to enable rule extraction for all handbooks, but to develop and demonstrate a framework that would be customizable. This aligns with the industry expectations of a framework that would then be customized as per its internally authored rules. Among several thousands of the sentences, 133 sentences are chosen to create the dataset. The following guideline is used to choose the sentences with rule-like information.

- The sentences specifying the constraint between the features are regarded as rules.

  e.g. 1. "The entrance surface of a drilled hole should be perpendicular to its axis"

  e.g. 2. "For steel, the depth of end-milled slots should not be deeper than the cutter diameter"

- The sentences specifying the clear preference to a design over another design are regarded as rules.

  e.g. "Spotfacing is more economical than face milling for small, flat surfaces"

- The sentences specifying the recommendation without definite constraint or preference are not regarded as rules.

e.g. "The part design should be as simple as possible": Discarded since the desired degree of simple is not clear.

Then when standard NLP techniques cannot correctly analyze some of the sentences, the sentences were manually modified as follows: 1) sentence structures were modified when the subject or object of the sentences could not be derived correctly, such as the sentence "The minimum wall thickness for extrusions should be 0.4 mm." has been modified to "For extrusion, the minimum wall thickness should be 0.4 mm." as "extrusions" was wrongly regarded as the subject, 2) some of the terms/phrases were modified when standard NLP techniques frequently misinterpret them, such as the phrase "be greater than" has been modified to "exceed" when the phrase is not correctly interpreted, 3) when the sentences need too much modification, the sentences were discarded.

Furthermore, some of the sentences were manually modified such that the rules are precise to focus on domain concepts, e.g., removing phrases such as "off the shelf" and replacing with the phrase "standard"; or that the rules align with the manufacturing domain ontology, e.g., remove phrases such as "corner shapes" from the sentence "Slot widths, radii, chamfers, corner shapes, and overall forms should conform to ..." because the manufacturing domain ontology does not capture the concept of "corner shape". Table A.1 shows all the 133 sentences in the dataset and the original sentences in the handbook.

Table A.1: The list of sentences in the dataset and original sentences

| Id | Original sentence | Sentence in the dataset |
|---|---|---|
| 1* | Sharp corners should be avoided because they interfere with smooth metal flow | Sharp corners should be avoided because they interfere with metal flow. |
| 2* | When keys and keyways are required, the desirable ratio of width to depth for ease of casting is 1.0 or more (see Fig. 5.3.7). | The width of a slot should exceed its depth. |
| 3* | Sharp edges and vertical sides also should be avoided, as shown in Fig. 8.5.7. | Sharp corners and edges should be avoided. |
| 4* | Sharp corners, both internal and external, should be avoided whenever possible. | It is better not to have sharp corners on a formed part. |
| 5* | However, if a blind hole is required, the length of depth should not exceed the diameter. | If a blind hole is required, its length of depth should not exceed its diameter. |
| 6 | If a sharp internal corner is necessary, the included angle should be as large as possible and always more than 90°. | If a sharp internal corner is necessary, the included angle should be always more than 90. |
| 7 | Sharp corners should be avoided because they interfere with smooth metal flow. | Avoid sharp edges of parts because they interfere with the smooth flow of material through the die. |
| 8 | In steel extrusions, the depth of an indentation should be no greater than its width at its narrowest point. | In steel extrusions, the depth of an indentation should not be greater than the width of narrow section of the part. |
| 9 | The ratio of length to thickness of any segment should not exceed 14:1. | The ratio of length to thickness of any segment should not exceed 14. |
| 10 | Symmetrical cross sections are preferable to nonsymmetrical designs to avoid unbalanced stresses and warpage. | Symmetrical cross sections are preferable to nonsymmetrical cross sections to avoid unbalanced stresses and warpage. |
| 11 | The diameter of pierced holes should be not less than stock thickness, as illustrated in Fig. 3.2.16. | The diameter of pierced holes should be not less than stock thickness. |

* Simplified entries, used as the motivating examples

Table A.1 (cont')

| Id | Original sentence | Sentence in the dataset |
|---|---|---|
| 12 | Pierce or drill the holes after forming. This is more expensive but provides excellent alignment. | Holes should not be pierced for a part to be formed. |
| 13 | As a general rule, long sections should not be narrower than 112 times stock thickness. | Width of long sections should not be smaller than 1.5 times stock thickness. |
| 14 | Tooling costs for round-hole punches and dies are far below those for holes of other than round shapes. | Punches for round holes are more economical than those for square holes. |
| 15 | The inside bend angle preferably should be 90°. | The inside bend angle should be 90. |
| 16 | Avoid tapered-wall shells and/or flanged shells. | Avoid tapered-walled shells or flanged shells. |
| 17 | On fine-blanked parts, corners must be rounded. | On fine-blanked parts, corners must be rounded. |
| 18 | Sharply defined contours and reentrant angles should be avoided. | Avoid reentrant shapes, because they add to the part cost. |
| 19 | Avoid reverse-form designs, if possible, because they require additional operations and can cause considerable thinning of stock. | Avoid reverse-formed designs, because they have additional operations and can cause considerable thinning of stock. |
| 20 | Hollow upsets should be avoided when possible because they require more die maintenance and there is a possibility of cracks forming around the edges of the recesses. | Hollow upsets should be avoided because they have more die maintenance and there is a possibility of cracks forming around the edges of the recesses. |
| 21 | In flat-headed parts, the slot must have a curved (instead of a flat) bottom to eliminate contact between the slotting punch and the dies. | In flat-headed parts, the slot must have a curved bottom to eliminate contact between the slotting punch and the dies. |
| 22 | Impact-extruded parts should be symmetrical to avoid lateral movement of tooling punches and unequal wall thickness. | Impact-extruded parts should be symmetrical to avoid lateral movement of tooling punches and unequal wall thickness. |

* Simplified entries, used as the motivating examples

Table A.1 (cont')

| Id | Original sentence | Sentence in the dataset |
|---|---|---|
| 23 | Spherical bottoms should be avoided, especially in long shapes; use an angular bottom instead, as shown in Fig. 3.8.8d. | Spherical bottoms should be avoided. |
| 24 | External ribs can have heights up to 2 times the wall thickness, but 1 times wall thickness is preferable. | The maximum height of external ribs is 2 times wall thickness. |
| 25 | The maximum recommended height of internal ribs is 3 times wall thickness. | The maximum height of internal ribs is 3 times wall thickness. |
| 26 | The diameter of an inside boss should be not more than one-fourth the diameter of the shell. | The diameter of an inside boss should not exceed 0.25 times the diameter of the shell. |
| 27 | Flanges should be symmetrical. | Flanges should be symmetrical. |
| 28 | They should have thicknesses equal to or greater than the sidewall. (See Fig. 3.8.13a.) | Thickness of flanges should be greater than that of the sidewall. |
| 29 | When swaging with shaped mandrels, the maximum special angle of flutes or grooves from the longitudinal axis is 30°. | When swaging with shaped mandrels, the maximum special angle of flutes is 30. |
| 30 | Minimum wall thickness T under normal conditions is 1.5 mm (0.060 in). | The minimum wall thickness is 1.5. |
| 31 | The normal maximum ratio of wall thickness to length is 18:1. | The maximum ratio of wall thickness to length is 18. |
| 32 | The normal maximum length of P/M parts is 100 mm (4 in). (See Fig. 3.12.6.) | The maximum length of P/M parts is 100. |
| 33 | The minimum diameter of holes is 1.5 mm (0.060 in). | The minimum diameter of holes is 1.5. |
| 34 | Avoid sharp internal corners on electroformed parts. | Avoid sharp internal corners on electroformed parts. |
| 35 | Inside corners should be well rounded to ensure an even deposit of metal. | Inside corners should be rounded to ensure an even deposit of metal. |

* Simplified entries, used as the motivating examples

Table A.1 (cont')

| Id | Original sentence | Sentence in the dataset |
|---|---|---|
| 36 | Undercuts, reverse tapers, and reentrant angles especially should be avoided, if possible, because they necessitate a flexible or dissolvable mandrel that is less accurate and more costly. | Undercuts, negative tapers, and reentrant angles especially should be avoided, because they necessitate a flexible or dissolvable mandrel that is less accurate and more costly. |
| 37 | Avoid undercuts, if possible, because they usually involve separate operations of specially ground tools. (See Fig. 4.1.5.) | Avoid undercuts because they usually involve separate operations of specially ground tools. |
| 38 | Sidewalls of grooves and other surfaces that are perpendicular to the axis of the workpiece should have a slight draft. | Draft of sidewalls of grooves that are perpendicular to the axis of the workpiece should be 0.5 or more. |
| 39 | A knurled area should be kept narrow. Its width should not exceed its diameter. | Width of a knurl should not exceed its diameter. |
| 40 | Avoid sharp corners in the design of screw-machine parts. | Avoid sharp corners in screw-machined parts. |
| 41 | A product design that requires an irregular and interrupted cutting action should be avoided when possible. | An irregular and interrupted cutting should be avoided. |
| 42 | Avoid sharp corners. | Avoid sharp corners. |
| 43 | The exit surface of the drill also should be perpendicular to the axis of the drill to avoid breakage problems as the drill leaves the work. | The exit surface of a hole also should be perpendicular to the axis of the drill to avoid breakage problems as the drill leaves the work. |
| 44 | When blind holes are specified, they should not have flat bottoms. | When blind holes are specified, they should not have flat bottoms. |
| 45 | Avoid deep holes (over 3 times diameter) because of chip-clearance problems and the possibility of deviations from straightness (see Fig. 4.5.10). | Avoid deep holes because of chip-clearance problems and the possibility of deviations from straightness. |

* Simplified entries, used as the motivating examples

Table A.1 (cont')

| Id | Original sentence | Sentence in the dataset |
|---|---|---|
| 46 | Avoid intersecting drilled and reamed holes if possible to prevent tool breakage and burr-removal problems. (See Fig. 4.5.8.) | Avoid reamed holes. |
| 47 | Drilled holes should not be deeper than 3 times diameter; bored holes can be as deep as 5 imes diameter. | Depth of a drilled hole should not exceed 3 times its diameter. |
| 48 | The entrance and exit surface should be perpendicular to the drill bit. | The entrance surface of a drilled hole should be perpendicular to its axis. |
| 49 | When a small, flat surface is required, as for a bearing surface or a bolt-head seat perpendicular to a hole, the product design should permit the use of spotfacing, which is quicker and more economical than face milling. | When a small, flat surface is perpendicular to an entrance surface of a hole, spotfacing is more economical than face milling. |
| 50 | For example, sharp inside and outside corners should be avoided. | Inside and outside corners of a part should not be sharp. |
| 51 | Slot widths, radii, chamfers, corner shapes, and overall forms should conform to those of cutters available off the shelf rather than those which require special fabrication. | Slot widths, radii, and chamfers should conform to those of standard cutters. |
| 52 | End-milled slots in steel normally should not be deeper than the cutter diameter. | For steel, the depth of end-milled slots should not be deeper than the cutter diameter. |
| 53 | A design that requires the milling of surfaces adjacent to a shoulder or flange should provide clearance for the cutter path. | When surfaces adjacent to a shoulder are milled, the design should have clearance for the cutter path. |
| 54 | Small steps or radii or inclined flange or shoulder surfaces as shown in Fig. 4.6.12 should be used. | The part should have small steps or radii or inclined flange or shoulder surfaces. |
| 55 | It is better not to specify a blended radius on machined rails. | It is better not to specify a blended radius on machined rails. |

* Simplified entries, used as the motivating examples

Table A.1 (cont')

| Id | Original sentence | Sentence in the dataset |
|---|---|---|
| 56 | In end-milling slots in mild steel, the depth should not exceed the diameter of the cutter. | For mild steel, the depth of end-milled slots should not exceed the diameter of the cutter. |
| 57 | Avoid multiple surfaces that are not parallel in the direction of reciprocating motion of the cutting tool because this would necessitate additional setups. | Avoid surfaces that are not parallel to cutter path of the cutting tool because this would necessitate additional setups. |
| 58 | Slots and contours should not be longer than 4 times the largest dimension of the opening or the hole diameter. | Length of slots and contours should not exceed 4 times the hole diameter. |
| 59 | Taper and flatness also should be avoided because they will not be removed by the thread-grinding operation. | Taper also should be avoided because they will not be removed by the thread-grinding. |
| 60 | Surfaces contiguous to the area to be cut should be square and relatively flat. | Surfaces adjacent to the area to be cut should be square and flat. |
| 61 | Long holes should be chambered as shown in Fig. 4.9.5 to improve accuracy as well as to reduce costs. | Long holes should be chambered to improve accuracy as well as to reduce costs. |
| 62 | Avoid dovetail or inverted-angle splines. | Dovetail or inverted-angle splines should be avoided. |
| 63 | Tapered splines should be avoided (see Fig. 4.9.11): | Tapered splines should be avoided. |
| 64 | Broaching blind holes should be avoided if at all possible. | Broaching blind holes should be avoided. |
| 65 | Sharp internal corners should be avoided to eliminate stress points and minimize tooth-edge wear. | Sharp internal corners should be avoided to eliminate stress points and minimize tooth-edge wear. |
| 66 | Sharp corners or edges of intersecting outer broached surfaces should be avoided whenever possible. | Sharp corners of broached outer surfaces should be avoided. |
| 67 | Contour-sawed holes should be avoided if possible. | Contour-sawed holes should be avoided. |

* Simplified entries, used as the motivating examples

Table A.1 (cont')

| Id | Original sentence | Sentence in the dataset |
|---|---|---|
| 68 | Flat-ground bottoms should be avoided. | Flat-grinding bottoms should be avoided. |
| 69 | The width of a centerless-ground part should be at least as large as its diameter. | When centerless-grinding a part, its width should be greater than its diameter. |
| 70 | Through holes are preferable to blind holes if roller burnishing is to be performed. | Through holes are preferable to blind holes if roller burnishing is performed. |
| 71 | Heat-treated gears should be of uniform cross section to minimize heat-treatment distortion. | Heat-treated gears should have uniform cross section to minimize heat-treatment distortion. |
| 72 | To facilitate removal, the pattern must have some degree of taper, or draft. | To facilitate removal, the pattern must have taper or draft. |
| 73 | The taper of the wedged area should not exceed 1:4. | The taper of the wedged area should not exceed 0.25. |
| 74 | Avoid sharp corners and fins in areas to be machined after casting. | Avoid sharp fins to be machined after casting. |
| 75 | A stock allowance must be added to surfaces that are to be machined. | Surfaces to be machined must have stock allowance. |
| 76 | Keys should not be narrower than 2.3 mm (0.090 in) for ferrous metals and 1.5 mm (0.060 in) for nonferrous metals. | For ferrous metals, width of keys should not be smaller than 2.3. |
| 77 | Keys should not be narrower than 2.3 mm (0.090 in) for ferrous metals and 1.5 mm (0.060 in) for nonferrous metals. | For nonferrous alloys, the minimum width of a castable key is 1.5. |
| 78 | However, if a blind hole is required, the length of depth should not exceed the diameter. | However, if a blind hole is required, the length of depth should not exceed the diameter. |
| 79 | To avoid sinks, ribs should be no wider than the thickness of the casting wall. | To avoid sinks, the width of ribs should not be greater than the thickness of the casting wall. |

* Simplified entries, used as the motivating examples

Table A.1 (cont')

| Id | Original sentence | Sentence in the dataset |
|---|---|---|
| 80 | A lower limit on core diameter of 3 mm (0.120 in) for aluminum and magnesium and 1.5 mm (0.060 in) for zinc should be observed, since smaller cores are prone to frequent breakage and erosion from heat buildup. | Minimum core diameter should be 3 for aluminum, since smaller cores are prone to frequent breakage and erosion from heat buildup. |
| 81 | Internal undercuts in a part are almost impossible to mold and should be avoided. (See Fig. 6.1.6.) | Internal undercuts in a part should be avoided. |
| 82 | Blind-hole depths, especially when the compression-molding process is used, should not be more than 212 times the diameter. | When the compression-molding is used, blind-hole depths should not exceed the diameter. |
| 83 | Side holes should be avoided since they require activated side cores. See Fig. 5.2.4 | Casting side holes should be avoided since they have activated side cores. |
| 84 | The width of the base of the rib should be less than the thickness of the wall to which it is attached. | The width of the base of the rib should be less than the thickness of the wall. |
| 85 | The boss height should not be more than twice the diameter. | The boss height should not be more than 2 times the diameter. |
| 86 | Blind holes should not be more than two diameters deep. | The depth of a blind hole should not exceed 2 times its diameter. |
| 87 | If the diameter is 1.5 mm (116 in) or less, one diameter is the maximum practical depth. (See Fig. 6.2.11.) | For a blind hole, If the diameter is smaller than 1.5, the maximum depth should be equal to the diameter. |
| 88 | Ribs should be perpendicular to the parting line to permit removal of the part from the mold. | Ribs should be perpendicular to the parting line to permit removal of the part from the mold. |
| 89 | As shown in j, the depth of insertion should be at least 2 times the insert diameter. | The depth of insertion should exceed 2 times its diameter. |

* Simplified entries, used as the motivating examples

Table A.1 (cont')

| Id | Original sentence | Sentence in the dataset |
|---|---|---|
| 90 | If the outside diameter of the insert is less than 6 mm (14 in), the outside diameter of the boss should be twice that of the insert. | If the outside diameter of an insert is smaller than 6, it should be 0.5 times the outside diameter of the boss. |
| 91 | If the diameter of the insert is 14 in or less, the boss diameter should be at least twice that of the insert. | If the diameter of an insert is 6.25 or less, the boss diameter should exceed 2 times the diameter of the insert. |
| 92 | Bosses and other thick sections should be cored. | Bosses and thick sections should be cored. |
| 93 | Connecting ribs should be of the same thickness as the part walls. | Thickness of connecting ribs should be equal to the thickness of the part walls. |
| 94 | Inside or outside corners should be well radiused and not sharp. | Inside or outside corners should be well radiused. |
| 95 | Avoid narrow projections or recesses in rotomolded parts. | Avoid narrow projections or recesses in rotomolded parts. |
| 96 | The minimum recommended draft angle is 12°. | The minimum draft angle is 0.5. |
| 97 | Press Molding. Undercuts should be avoided. | For press molding, undercuts should be avoided. |
| 98 | The width of the shoulder should be no larger than the diameter of the hole. | The width of the shoulder should not be greater than the diameter of the hole. |
| 99 | Gently curved or rounded surfaces are preferred to flat surfaces. | Curved or rounded surfaces are preferred to flat surfaces. |
| 100 | Sharp inside corners also should be avoided because they can form a notch that can be an easy breaking point for the more rigid plastics. | Sharp inside corners also should be avoided because they can form a notch that can be an easy breaking point for the more rigid plastics. |
| 101 | As a general rule, corner radii should be at least one-half the wall thickness. | Corner radii should exceed 1.5 times the wall thickness. |
| 102 | For straight vacuum forming into a female mold, the depth-to-width ratio should not exceed 0.5:1. | For straight vacuum forming with a female mold, the depth-to-width ratio should not exceed 0.5. |

* Simplified entries, used as the motivating examples

Table A.1 (cont')

| Id | Original sentence | Sentence in the dataset |
|---|---|---|
| 103 | For drape forming over a male mold, the depth-to-width ratio should not exceed 1:1. | For drape forming with a male mold, the depth-to-width ratio should not exceed 1. |
| 104 | Deep undercuts should be avoided; shallow undercuts sometimes can be sprung from the mold when the workpiece is still warm. | Deep undercuts should be avoided. |
| 105 | Through holes are preferable to blind holes because the pins forming them can be anchored at both ends and are therefore less susceptible to deflection during molding. | Through holes are preferable to blind holes because the pins forming them can be anchored at both ends and are therefore less susceptible to deflection during molding. |
| 106 | Undercuts should be avoided if possible. | Undercuts should be avoided. |
| 107 | The minimum radius should be at least 1.5 times the outside diameter, as shown in Fig. 6.10.22. | The minimum radius should exceed 1.5 times the outside diameter. |
| 108 | Avoid featheredges. | Avoid featheredges. |
| 109 | When parts are machined, outside radii should be 1.5 mm (116 in) or more and inside radii at least 2.4 mm (332 in). | When a part is machined, the outside radii should be greater than 1.5. |
| 110 | For dry-pressed parts, outside edges should be beveled in a manner similar to that employed with powder-metal parts; 0.8 mm by 45°is a desirable minimum. | For dry-pressed parts, outside edges should be beveled similar to that employed with powder-metal parts. |
| 111 | Sections should not exceed 25 mm (1 in) in thickness. (See Table 6.11.3 for wall-thickness information.) | Thickness of sections should not exceed 25. |
| 112 | Undercuts should be avoided in ceramic components if possible. | Undercuts should be avoided in ceramic parts. |
| 113 | The minimum wall thickness for extrusions should be 0.4 mm (164 in). | For extrusion, the minimum wall thickness should be 0.4. |

* Simplified entries, used as the motivating examples

Table A.1 (cont')

| Id | Original sentence | Sentence in the dataset |
|---|---|---|
| 114 | Machined holes should be at least 1.5 mm (116 in) in diameter if possible, although smaller holes can be produced. | Diameter of machined holes should exceed 1.5 , although smaller holes can be produced. |
| 115 | All tapped holes should be countersunk. (See Fig. 6.11.9.) | All tapped holes should be countersunk. |
| 116 | Ribs and fins should be well rounded, wide, and well spaced and have normal draft. | Ribs and fins should be well rounded. |
| 117 | Deep, narrow joint areas should be avoided. | Deep, narrow joint areas should be avoided. |
| 118 | With aluminum, the maximum recommended stack height is 8 mm (516 in) and with magnesium 6.3 mm (14 in). | With aluminum, the maximum stack height is 8. |
| 119 | Bars over 6 mm in diameter and tubing with a wall thickness over 5 mm should be beveled as shown before flash welding. | If tube wall thickness exceeds 5, it should be beveled before flash welding. |
| 120 | Angle welds should be avoided if possible because the upsetting force is not applied so squarely to the joint. | Angle welds should be avoided because the upsetting force is not applied so squarely to the joint. |
| 121 | For tubing and other sections, the joint angle should not normally be less than 150°. | For tubing, the joint angle should not be smaller than 150. |
| 122 | A generous radius should be allowed at all inside corners. | All internal corners should be rounded. |
| 123 | Avoid sharp internal corners, which concentrate heat-treating stresses. | Avoid sharp internal corners, which concentrate heat-treating stresses. |
| 124 | Avoid part designs that require paint masking. | Avoid part designs that require paint masking. |
| 125 | Avoid sharp edges and points, whenever possible, because paint tends to creep away from these areas and leave too thin a film. | Avoid sharp edges and points, because paint tends to creep away from these areas and leave too thin a film. |
| 126 | Avoid blind areas, i.e., surfaces that are behind corners or protrusions. | Avoid blind areas, i.e., surfaces that are behind corners or protrusions. |

* Simplified entries, used as the motivating examples

Table A.1 (cont')

| Id | Original sentence | Sentence in the dataset |
|---|---|---|
| 127 | Deep recesses, however, should be avoided, particularly if they are narrow. | Deep recesses, however, should be avoided, particularly if they are narrow. |
| 128 | Parts to be curtain-coated should be flat and without depressions. | Curtain-coated parts should not have depressions. |
| 129 | Sharp edges and vertical sides also should be avoided, as shown in Fig. 8.5.7. | Sharp edges and vertical sides also should be avoided. |
| 130 | If an electrostatic powder-spray process is used, sharp edges and recesses should be avoided. | If an electrostatic powder-spraying is used, sharp edges and recesses should be avoided. |
| 131 | Overhangs and deep and narrow recesses should be avoided. | Overhangs and deep and narrow recesses should be avoided. |
| 132 | The stroke (letter leg width) should be at least three times the material thickness. | The stroke should exceed 3 times the stock thickness. |
| 133 | Avoid sharp internal corners. | Avoid sharp internal corners. |

* Simplified entries, used as the motivating examples

## APPENDIX B: THE LIST OF GRAMMATICAL DEPENDENCIES

In this dissertation, the grammatical dependencies between natural language terms are derived by NLP4J [79]. Table B.1 shows the list of dependencies.

Table B.1: The list of grammatical dependencies used in this dissertation

| Label | Description | Label | Description |
|---|---|---|---|
| acl | Clausal modifier of noun | mark | Marker |
| acomp | Adjectival complement | meta | Meta data |
| advcl | Adverbial clause modifier | neg | Negation modifier |
| advmod | Adverbial modifier | nmod | Modifier of nominal |
| agent | Agent (passive) | npadvmod | Noun phrase as adverbial modifier |
| appos | Appositional modifier | nsubj | Nominal subject |
| attr | Attribute | nsubjpass | Nominal subject (passive) |
| aux | Auxiliary verb | oprd | Object predicate |
| auxpass | Auxiliary verb (passive) | parataxis | Parataxis |
| case | Case marker | pcomp | Preposition complement |
| cc | Coordinating conjunction | pobj | Preposition object |
| ccomp | Clausal complement | poss | Possession modifier |
| compound | Compound word | preconj | Precorrelative conjunction |
| conj | Conjunct | predet | Predeterminer |
| coref | Co-reference | prep | Prepositional modifier |
| csubj | Clausal subject | prt | Verb particle |
| csubjpass | Clausal subject (passive) | punct | Punctuation |
| dative | Dative | qmod | Modifier of quantifier |
| dep | Unclassified dependent | relcl | Relative clause modifier |
| det | Determiner | root | Root |
| discourse | Discourse element | vocative | Vocative modifier |
| dobj | Direct Object | xcomp | Open clausal complement |
| expl | Expletive | | |

# APPENDIX C: THE SUMMARY OF RULE EXTRACTION RESULTS

Table C.1 shows the summary of the rule extraction results. Specifically, there are three cases of the results as follows: 1) correct rule was extracted using the rule extraction framework, 2) correct rule was extracted after extending the framework with ambiguity resolution method, and 3) correct rule was not extracted as it requires the input text validation. For the details about the extracted rules, the readers are referred to Appendix D, Appendix E, and Appendix F. First, all the rules extracted with the basic rule extraction framework (Chapter 2) are shown in Appendix D. For the cases that correct rules could be extracted after adopting the ambiguity resolution method (Chapter 3), the correctly extracted rules are shown in Appendix E. For the cases that needed input text validation, the input sentences were manually modified based on the response from the feedback generation method (Chapter 4). Following that, the rules were correctly extracted as shown in Appendix F.

Table C.1: The summary of rule extraction results

| Id | Correct rule extracted by rule extraction framework (Chapter 2) | Correct rule extracted after ambiguity resolution (Chapter 3) | Correct rule not extracted (Chapter 4) |
|---|---|---|---|
| 1 | | | X |
| 2 | | X | |
| 3 | | X | |
| 4 | | | X |
| 5 | | | X |
| 6 | | | X |
| 7 | | | X |
| 8 | X | | |
| 9 | X | | |
| 10 | | | X |

Table C.1 (cont')

| Id | Correct rule extracted by rule extraction framework (Chapter 2) | Correct rule extracted after ambiguity resolution (Chapter 3) | Correct rule not extracted (Chapter 4) |
|---|---|---|---|
| 11 | X | | |
| 12 | X | | |
| 13 | X | | |
| 14 | | X | |
| 15 | | | X |
| 16 | X | | |
| 17 | X | | |
| 18 | X | | |
| 19 | | | X |
| 20 | | | X |
| 21 | | | X |
| 22 | | | X |
| 23 | X | | |
| 24 | X | | |
| 25 | X | | |
| 26 | X | | |
| 27 | X | | |
| 28 | | X | |
| 29 | X | | |
| 30 | X | | |
| 31 | X | | |
| 32 | X | | |
| 33 | X | | |
| 34 | X | | |
| 35 | | | X |
| 36 | | | X |
| 37 | | | X |
| 38 | X | | |
| 39 | | X | |
| 40 | X | | |
| 41 | X | | |
| 42 | X | | |
| 43 | | | X |
| 44 | | X | |
| 45 | X | | |
| 46 | X | | |

Table C.1 (cont')

| Id | Correct rule extracted by rule extraction framework (Chapter 2) | Correct rule extracted after ambiguity resolution (Chapter 3) | Correct rule not extracted (Chapter 4) |
|---|---|---|---|
| 47 | | X | |
| 48 | | X | |
| 49 | X | | |
| 50 | | X | |
| 51 | | X | |
| 52 | X | | |
| 53 | X | | |
| 54 | X | | |
| 55 | | | X |
| 56 | X | | |
| 57 | X | | |
| 58 | X | | |
| 59 | | | X |
| 60 | X | | |
| 61 | X | | |
| 62 | X | | |
| 63 | X | | |
| 64 | X | | |
| 65 | | | X |
| 66 | X | | |
| 67 | X | | |
| 68 | X | | |
| 69 | | X | |
| 70 | X | | |
| 71 | X | | |
| 72 | X | | |
| 73 | X | | |
| 74 | X | | |
| 75 | X | | |
| 76 | X | | |
| 77 | X | | |
| 78 | | | X |
| 79 | | | X |
| 80 | | | X |
| 81 | X | | |
| 82 | | | X |

Table C.1 (cont')

| Id | Correct rule extracted by rule extraction framework (Chapter 2) | Correct rule extracted after ambiguity resolution (Chapter 3) | Correct rule not extracted (Chapter 4) |
|---|---|---|---|
| 83 | | | X |
| 84 | X | | |
| 85 | | | X |
| 86 | | X | |
| 87 | | | X |
| 88 | | | X |
| 89 | | X | |
| 90 | X | | |
| 91 | X | | |
| 92 | X | | |
| 93 | X | | |
| 94 | | X | |
| 95 | | X | |
| 96 | | | X |
| 97 | X | | |
| 98 | X | | |
| 99 | X | | |
| 100 | | | X |
| 101 | X | | |
| 102 | | | X |
| 103 | | | X |
| 104 | X | | |
| 105 | | | X |
| 106 | X | | |
| 107 | | | X |
| 108 | X | | |
| 109 | | | X |
| 110 | | | X |
| 111 | X | | |
| 112 | X | | |
| 113 | X | | |
| 114 | | | X |
| 115 | X | | |
| 116 | X | | |
| 117 | X | | |
| 118 | X | | |

Table C.1 (cont')

| Id | Correct rule extracted by rule extraction framework (Chapter 2) | Correct rule extracted after ambiguity resolution (Chapter 3) | Correct rule not extracted (Chapter 4) |
|---|---|---|---|
| 119 | | X | |
| 120 | X | | |
| 121 | | | X |
| 122 | X | | |
| 123 | | | X |
| 124 | X | | |
| 125 | | | X |
| 126 | | | X |
| 127 | X | | |
| 128 | X | | |
| 129 | X | | |
| 130 | | X | |
| 131 | X | | |
| 132 | | | X |
| 133 | X | | |

| Total rules | Correct rule extracted by rule extraction framework (Chapter 2) | Correct rule extracted after ambiguity resolution (Chapter 3) | Correct rule not extracted (Chapter 4) |
|---|---|---|---|
| 133 | 76 | 93 | 40 |

## APPENDIX D: THE DETAILS OF RULE EXTRACTION RESULTS

Table D.1 shows the full list of the rules extracted with basic rule extraction framework (Chapter 2).

Table D.1: The list of the rules extracted with basic rule extraction framework (Chapter 2)

| Id | Extracted rule |
|---|---|
| 1 | Given<br>(design_id0 is a MechnicalDesignService)<br>(metalFlow_id11 is a MetalFlow)<br><br>If<br>(corner_id2 interfereWith metalFlow_id11)<br>(corner_id2 isSharp true)<br>(metalFlow_id11 isSmooth true)<br>( (corner_id2 is a Corner)<br>and (corner_id2 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback1 isNegative true<br>feedback1 has comment “Sharp corners should be avoided because they interfere with smooth metal flow.”. |
| 2 | Given<br>(design_id0 is a MechnicalDesignService)<br>(slot_id5 is a Slot)<br>and (slot_id5 is hasFeature of hasPart of design_id0)<br>If<br>not (hasWidth of slot_id5 > hasDepth of null)<br><br>Then<br>feedback2 isNegative true<br>feedback2 has comment “The width of a slot should exceed its depth.”. |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 3 | Given<br>(design_id0 is a MechnicalDesignService)<br><br>If<br>(corner_id2 isSharp true)<br>(<br>( (edge_id4 is a Edge)<br>and (edge_id4 is hasFeature of hasPart of design_id0) ) or<br>( (corner_id2 is a Corner)<br>and (corner_id2 is hasFeature of hasPart of design_id0) )<br>)<br><br>Then<br>feedback3 isNegative true<br>feedback3 has comment "Sharp corners and edges should be avoided .". |
| 4 | Given<br>(design_id0 is a MechnicalDesignService)<br>(forming_id11 is a Forming)<br>and (forming_id11 is hasProcess of design_id0)<br>(part_id13 is a Part)<br>and (part_id13 is hasTargetPart of forming_id11)<br>and (part_id13 is hasPart of design_id0)<br>(corner_id8 is a Corner)<br><br>If<br>not (null have corner_id8)<br>(corner_id8 isSharp true)<br><br>Then<br>feedback4 isNegative true<br>feedback4 has comment "It is better not to have sharp corners on a formed part.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 5 | Given<br>(design_id0 is a MechnicalDesignService)<br>(blindHole_id3 is a BlindHole)<br>and (blindHole_id3 is hasFeature of hasPart of design_id0)<br><br>If<br>(hasDepth of null > 0)<br>(hasLength of null > hasDiameter of null)<br><br>Then<br>feedback5 isNegative true<br>feedback5 has comment "If a blind hole is required, its length of depth should not exceed its diameter.". |
| 6 | Given<br>(design_id0 is a MechnicalDesignService)<br>(extruding_id2 is a Extruding)<br>and (extruding_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of extruding_id2)<br>(internalCorner_id6 is a InternalCorner)<br>and (internalCorner_id6 is hasFeature of hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(internalCorner_id6 isSharp true)<br>not (hasIncludedAngle of null > 90)<br><br>Then<br>feedback6 isNegative true<br>feedback6 has comment "If a sharp internal corner is necessary, the included angle should be always more than 90.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 7 | Given<br>(design_id0 is a MechnicalDesignService)<br>(extruding_id1 is a Extruding)<br>and (extruding_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of extruding_id1)<br>(die_id20 is a Die)<br>and (die_id20 is hasTool of extruding_id1)<br>(part_id7 is a Part)<br>and (part_id7 is hasPart of design_id0)<br>(material_id17 is a Material)<br>and (part_id7 isMadeOf material_id17)<br>(flow_id14 is a Flow)<br><br>If<br>(part_id7 hasEdge edge_id4)<br>(edge_id4 interfereWith flow_id14)<br>(edge_id4 isSharp true)<br>(flow_id14 isSmooth true)<br>( (edge_id4 is a Edge)<br>and (edge_id4 is hasEdge of part_id7) )<br><br>Then<br>feedback7 isNegative true<br>feedback7 has comment "Avoid sharp edges of parts because they interfere with the smooth flow of material through the die.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 8 | Given<br>(design_id0 is a MechnicalDesignService)<br>(steelExtruding_id3 is a SteelExtruding)<br>and (steelExtruding_id3 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of steelExtruding_id3)<br>(part_id25 is a Part)<br>and (part_id25 is hasPart of design_id0)<br>(indentation_id11 is a Indentation)<br>and (indentation_id11 is hasFeature of part_id25)<br>(section_id21 is a Section)<br>and (section_id21 is hasSection of part_id25)<br>(metal_id1 is a Metal)<br>and (part_id25 isMadeOf metal_id1)<br><br>If<br>(section_id21 isNarrow true)<br>(hasDepth of indentation_id11 > hasWidth of section_id21)<br><br>Then<br>feedback8 isNegative true<br>feedback8 has comment "In steel extrusions, the depth of an indentation should not be greater than the width of narrow section of the part.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 9 | Given<br>(design_id0 is a MechnicalDesignService)<br>(extruding_id2 is a Extruding)<br>and (extruding_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of extruding_id2)<br>(segment_id11 is a Segment)<br>and (segment_id11 is hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (segment_id11 isMadeOf metal_id1)<br><br>If<br>(hasLength of segment_id11 > 0)<br>(hasThickness of segment_id11 > 0)<br>(hasLength of segment_id11 / hasThickness of segment_id11 > 14)<br><br>Then<br>feedback9 isNegative true<br>feedback9 has comment "The ratio of length to thickness of any segment should not exceed 14.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 10 | Given<br>(design_id0 is a MechnicalDesignService)<br>(extruding_id2 is a Extruding)<br>and (extruding_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of extruding_id2)<br>(crossSection_id11 is a CrossSection)<br>and (crossSection_id11 is hasFeature of hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br>(warpage_id19 is a Warpage)<br>(stress_id17 is a Stress)<br><br>If<br>(crossSection_id11 isNonsymmetrical true)<br>(null avoid true)<br>(stress_id17 isUnbalanced true)<br><br>Then<br>feedback10 isNegative true<br>feedback10 has comment "Symmetrical cross sections are preferable to nonsymmetrical cross sections to avoid unbalanced stresses and warpage.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 11 | Given<br>(design_id0 is a MechnicalDesignService)<br>(piercing_id5 is a Piercing)<br>and (piercing_id5 is hasProcess of design_id0)<br>(hole_id7 is a Hole)<br>and (hole_id7 is hasTargetFeature of piercing_id5)<br>and (hole_id7 is hasFeature of hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br>(workpiece_id13 is a Workpiece)<br>and (workpiece_id13 is hasProcessInput of piercing_id5)<br><br>If<br>(hasDiameter of hole_id7 $<$ hasThickness of workpiece_id13)<br><br>Then<br>feedback11 isNegative true<br>feedback11 has comment "The diameter of pierced holes should be not less than stock thickness.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 12 | Given<br>(design_id0 is a MechnicalDesignService)<br>(forming_id14 is a Forming)<br>and (forming_id14 is hasProcess of design_id0)<br>(piercing_id6 is a Piercing)<br>and (piercing_id6 is hasProcess of design_id0)<br>(part_id11 is a Part)<br>and (part_id11 is hasTargetPart of piercing_id6)<br>and (part_id11 is hasTargetPart of forming_id14)<br>and (part_id11 is hasPart of design_id0)<br>(hole_id2 is a Hole)<br>and (hole_id2 is hasTargetFeature of piercing_id6)<br>and (hole_id2 is hasFeature of part_id11)<br>(metal_id1 is a Metal)<br>and (part_id11 isMadeOf metal_id1)<br><br>If<br>(forming_id14 toBePerformed true)<br><br>Then<br>feedback12 isNegative true<br>feedback12 has comment "Holes should not be pierced for a part to be formed.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 13 | Given<br>(design_id0 is a MechnicalDesignService)<br>(stamping_id2 is a Stamping)<br>and (stamping_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of stamping_id2)<br>(section_id6 is a Section)<br>and (section_id6 is hasFeature of hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br>(workpiece_id14 is a Workpiece)<br>and (workpiece_id14 is hasProcessInput of stamping_id2)<br><br>If<br>(section_id6 isLong true)<br>(hasWidth of section_id6 $<$ 1.5*hasThickness of workpiece_id14)<br><br>Then<br>feedback13 isNegative true<br>feedback13 has comment "Width of long sections should not be smaller than 1.5 times stock thickness.". |
| 14 | Given<br>(design_id0 is a MechnicalDesignService)<br>(stamping_id2 is a Stamping)<br>and (stamping_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of stamping_id2)<br>(hole_id16 is a Hole)<br>and (hole_id16 is hasFeature of hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(hole_id16 isSquare true)<br><br>Then<br>feedback14 isNegative true<br>feedback14 has comment "Punches for round holes are more economical than those for square holes.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 15 | Given<br>(design_id0 is a MechnicalDesignService)<br>(stamping_id2 is a Stamping)<br>and (stamping_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of stamping_id2)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>not (hasInsideBendAngle of null == 90)<br><br>Then<br>feedback15 isNegative true<br>feedback15 has comment "The internal bend angle should be 90.". |
| 16 | Given<br>(design_id0 is a MechnicalDesignService)<br>(stamping_id2 is a Stamping)<br>and (stamping_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of stamping_id2)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(shell_id7 isTapered-walled true)<br>(shell_id10 isFlanged true)<br>(<br>( (shell_id10 is a Shell)<br>and (shell_id10 is hasFeature of hasPart of design_id0) ) or<br>( (shell_id7 is a Shell)<br>and (shell_id7 is hasFeature of hasPart of design_id0) )<br>)<br><br>Then<br>feedback16 isNegative true<br>feedback16 has comment "Avoid tapered-walled shells or flanged shells.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 17 | Given<br>(design_id0 is a MechnicalDesignService)<br>(fine-blanking_id5 is a Fine-blanking)<br>and (fine-blanking_id5 is hasProcess of design_id0)<br>(part_id7 is a Part)<br>and (part_id7 is hasTargetPart of fine-blanking_id5)<br>and (part_id7 is hasPart of design_id0)<br>(corner_id9 is a Corner)<br>and (corner_id9 is hasFeature of part_id7)<br>(metal_id1 is a Metal)<br>and (part_id7 isMadeOf metal_id1)<br><br>If<br>not (corner_id9 isRounded true)<br><br>Then<br>feedback17 isNegative true<br>feedback17 has comment "On fine-blanked parts, corners must be rounded.". |
| 18 | Given<br>(design_id0 is a MechnicalDesignService)<br>(spinning_id2 is a Spinning)<br>and (spinning_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of spinning_id2)<br>(part_id12 is a Part)<br>and (part_id12 is hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (part_id12 isMadeOf metal_id1)<br><br>If<br>(shape_id5 isReentrant true)<br>( (shape_id5 is a Shape)<br>and (shape_id5 is hasGeometry of hasFeature of part_id12) )<br><br>Then<br>feedback18 isNegative true<br>feedback18 has comment "Avoid reentrant shapes, because they add to the part cost.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 19 | Given<br>(mfgProcess_id12 is a MfgProcess)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id6)<br>(workpiece_id19 is a Workpiece)<br>and (workpiece_id19 is hasProcessInput of mfgProcess_id12)<br><br>If<br>(mfgProcess_id12 is hasProcess of design_id6)<br>(hasPart of design_id6 is hasTargetPart of mfgProcess_id12)<br>(design_id6 isReverse-formed true)<br>( (design_id6 is a MechnicalDesignService) )<br><br>Then<br>feedback19 isNegative true<br>feedback19 has comment "Avoid reverse-formed designs, because they have additional operations and can cause considerable thinning of stock.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 20 | Given<br>(design_id0 is a MechnicalDesignService)<br>(forming_id20 is a Forming)<br>and (forming_id20 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of forming_id20)<br>(die_id11 is a Die)<br>and (die_id11 is hasTool of forming_id20)<br>(recess_id27 is a Recess)<br>and (recess_id27 is hasFeature of hasPart of design_id0)<br>(edge_id23 is a Edge)<br>and (edge_id23 is hasEdge of recess_id27)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(upset_id3 have maintenance_id12(Unclassified))<br>(upset_id3 isHollow true)<br>( (upset_id3 is a Upset)<br>and (upset_id3 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback20 isNegative true<br>feedback20 has comment "Hollow upsets should be avoided because they have more die maintenance and there is a possibility of cracks forming around the edges of the recesses.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 21 | Given<br>(design_id0 is a MechnicalDesignService)<br>(slotting_id20 is a Slotting)<br>and (slotting_id20 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of slotting_id20)<br>(punch_id21 is a Punch)<br>and (punch_id21 is hasTool of slotting_id20)<br>(part_id6 is a Part)<br>and (part_id6 is hasPart of design_id0)<br>(slot_id9 is a Slot)<br>and (slot_id9 is hasFeature of part_id6)<br>(bottomSurface_id14 is a BottomSurface)<br>(metal_id1 is a Metal)<br>and (part_id6 isMadeOf metal_id1)<br><br>If<br>(part_id6 isFlat-headed true)<br>(bottomSurface_id14 isCurved true)<br>not (bottomSurface_id14 is hasBottomSurface of slot_id9)<br><br>Then<br>feedback21 isNegative true<br>feedback21 has comment "In flat-headed parts, the slot must have a curved bottom to eliminate contact between the slotting punch and the dies.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 22 | Given<br>(design_id0 is a MechnicalDesignService)<br>(impactExtruding_id4 is a ImpactExtruding)<br> and (impactExtruding_id4 is hasProcess of design_id0)<br>(punch_id16 is a Punch)<br> and (punch_id16 is hasTool of impactExtruding_id4)<br>(part_id6 is a Part)<br> and (part_id6 is hasTargetPart of impactExtruding_id4)<br> and (part_id6 is hasPart of design_id0)<br>(wall_id19 is a Wall)<br> and (wall_id19 is hasFeature of part_id6)<br>(metal_id1 is a Metal)<br> and (part_id6 isMadeOf metal_id1)<br><br>If<br>(null avoid true)<br>not (part_id6 isSymmetrical true)<br><br>Then<br>feedback22 isNegative true<br>feedback22 has comment "Impact-extruded parts should be symmetrical to avoid lateral movement of tooling punches and unequal wall thickness.". |
| 23 | Given<br>(design_id0 is a MechnicalDesignService)<br>(extruding_id2 is a Extruding)<br> and (extruding_id2 is hasProcess of design_id0)<br> and (hasPart of design_id0 is hasTargetPart of extruding_id2)<br>(metal_id1 is a Metal)<br> and (metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(bottomSurface_id4 isSpherical true)<br>( (bottomSurface_id4 is a BottomSurface)<br> and (bottomSurface_id4 is hasGeometry of hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback23 isNegative true<br>feedback23 has comment "Spherical bottoms should be avoided.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 24 | Given<br>(design_id0 is a MechnicalDesignService)<br>(extruding_id2 is a Extruding)<br>and (extruding_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of extruding_id2)<br>(externalRib_id7 is a ExternalRib)<br>and (externalRib_id7 is hasFeature of hasPart of design_id0)<br>(wall_id13 is a Wall)<br>and (wall_id13 is hasFeature of hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>not (hasHeight of externalRib_id7 < 2*hasThickness of wall_id13)<br><br>Then<br>feedback24 isNegative true<br>feedback24 has comment "The maximum height of external ribs is 2 times wall thickness.". |
| 25 | Given<br>(design_id0 is a MechnicalDesignService)<br>(extruding_id2 is a Extruding)<br>and (extruding_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of extruding_id2)<br>(internalRib_id7 is a InternalRib)<br>and (internalRib_id7 is hasFeature of hasPart of design_id0)<br>(wall_id13 is a Wall)<br>and (wall_id13 is hasFeature of hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>not (hasHeight of internalRib_id7 < 3*hasThickness of wall_id13)<br><br>Then<br>feedback25 isNegative true<br>feedback25 has comment "The maximum height of internal ribs is 3 times wall thickness.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 26 | Given<br>(design_id0 is a MechnicalDesignService)<br>(extruding_id2 is a Extruding)<br>and (extruding_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of extruding_id2)<br>(internalBoss_id7 is a InternalBoss)<br>and (internalBoss_id7 is hasFeature of hasPart of design_id0)<br>(shell_id19 is a Shell)<br>and (shell_id19 is hasFeature of hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(hasDiameter of internalBoss_id7 > 0.25*hasDiameter of shell_id19)<br><br>Then<br>feedback26 isNegative true<br>feedback26 has comment "The diameter of an internal boss should not exceed 0.25 times the diameter of the shell.". |
| 27 | Given<br>(design_id0 is a MechnicalDesignService)<br>(extruding_id2 is a Extruding)<br>and (extruding_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of extruding_id2)<br>(flange_id3 is a Flange)<br>and (flange_id3 is hasFeature of hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>not (flange_id3 isSymmetrical true)<br><br>Then<br>feedback27 isNegative true<br>feedback27 has comment "Flanges should be symmetrical.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 28 | Given<br>(design_id0 is a MechnicalDesignService)<br>(extruding_id2 is a Extruding)<br>and (extruding_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of extruding_id2)<br>(flange_id6 is a Flange)<br>and (flange_id6 is hasFeature of hasPart of design_id0)<br>(sideWall_id15 is a SideWall)<br>and (sideWall_id15 is hasFeature of hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>not (hasThickness of flange_id6 > that(Unclassified))<br><br>Then<br>feedback28 isNegative true<br>feedback28 has comment "Thickness of the flange should be greater than that of the sidewall.". |
| 29 | Given<br>(design_id0 is a MechnicalDesignService)<br>(swaging_id3 is a Swaging)<br>and (swaging_id3 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of swaging_id3)<br>(shapedMandrel_id5 is a ShapedMandrel)<br>and (shapedMandrel_id5 is hasTool of swaging_id3)<br>(flute_id15 is a Flute)<br>and (flute_id15 is hasFeature of hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>not (hasSpecialAngle of flute_id15 < 30)<br><br>Then<br>feedback29 isNegative true<br>feedback29 has comment "When swaging with shaped mandrels, the maximum special angle of flutes is 30.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 30 | Given<br>(design_id0 is a MechnicalDesignService)<br>(metallurgy_id2 is a Metallurgy)<br>and (metallurgy_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of metallurgy_id2)<br>(wall_id5 is a Wall)<br>and (wall_id5 is hasFeature of hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>not (hasThickness of wall_id5 > 1.5)<br><br>Then<br>feedback30 isNegative true<br>feedback30 has comment "The minimum wall thickness is 1.5.". |
| 31 | Given<br>(design_id0 is a MechnicalDesignService)<br>(metallurgy_id2 is a Metallurgy)<br>and (metallurgy_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of metallurgy_id2)<br>(wall_id7 is a Wall)<br>and (wall_id7 is hasFeature of hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(hasThickness of wall_id7 > 0)<br>(hasLength of wall_id7 > 0)<br>not (hasThickness of wall_id7/hasLength of wall_id7 < 18)<br><br>Then<br>feedback31 isNegative true<br>feedback31 has comment "The maximum ratio of wall thickness to length is 18.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 32 | Given<br>(design_id0 is a MechnicalDesignService)<br>(powderMetallurgy_id6 is a PowderMetallurgy)<br>and (powderMetallurgy_id6 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of powderMetallurgy_id6)<br>(part_id9 is a Part)<br>and (part_id9 is hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (part_id9 isMadeOf metal_id1)<br><br>If<br>not (hasLength of part_id9 < 100)<br><br>Then<br>feedback32 isNegative true<br>feedback32 has comment "The maximum length of Powder metallurgy parts is 100.". |
| 33 | Given<br>(design_id0 is a MechnicalDesignService)<br>(metallurgy_id2 is a Metallurgy)<br>and (metallurgy_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of metallurgy_id2)<br>(hole_id7 is a Hole)<br>and (hole_id7 is hasFeature of hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>not (hasDiameter of hole_id7 > 1.5)<br><br>Then<br>feedback33 isNegative true<br>feedback33 has comment "The minimum diameter of holes is 1.5.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 34 | Given<br>(design_id0 is a MechnicalDesignService)<br>(electroforming_id8 is a Electroforming)<br>and (electroforming_id8 is hasProcess of design_id0)<br>(part_id10 is a Part)<br>and (part_id10 is hasTargetPart of electroforming_id8)<br>and (part_id10 is hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (part_id10 isMadeOf metal_id1)<br><br>If<br>(internalCorner_id4 isSharp true)<br>( (internalCorner_id4 is a InternalCorner)<br>and (internalCorner_id4 is hasFeature of part_id10) )<br><br>Then<br>feedback34 isNegative true<br>feedback34 has comment "Avoid sharp internal corners on electroformed parts.". |
| 35 | Given<br>(design_id0 is a MechnicalDesignService)<br>(electroforming_id1 is a Electroforming)<br>and (electroforming_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of electroforming_id1)<br>(internalCorner_id2 is a InternalCorner)<br>and (internalCorner_id2 is hasFeature of hasPart of design_id0)<br>(metal_id15 is a Metal)<br>and (metal_id15 is isMadeOf of hasPart of design_id0)<br>(deposit_id12 is a Deposit)<br><br>If<br>not (internalCorner_id2 isRounded true)<br><br>Then<br>feedback35 isNegative true<br>feedback35 has comment "Internal corners should be rounded to ensure an even deposit of metal.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 36 | Given<br>(design_id0 is a MechnicalDesignService)<br>(electroforming_id2 is a Electroforming)<br>and (electroforming_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of electroforming_id2)<br>(mandrel_id27 is a Mandrel)<br>and (mandrel_id27 is hasTool of electroforming_id2)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(<br>(hasReverseTaper of null $>$ 0) or<br>( (undercut_id3 is a Undercut)<br>and (undercut_id3 is hasFeature of hasPart of design_id0) ) or<br>(hasReentrantAngle of null $>$ 0)<br>)<br><br>Then<br>feedback36 isNegative true<br>feedback36 has comment "Relief grooves, negative tapers, and reentrant angles especially should be avoided, because they necessitate a flexible or dissolvable mandrel that is less accurate and more costly.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 37 | Given<br>(design_id0 is a MechnicalDesignService)<br>(grinding_id15 is a Grinding)<br> and (grinding_id15 is hasProcess of design_id0)<br> and (hasPart of design_id0 is hasTargetPart of grinding_id15)<br>(mfgProcess_id11 is a MfgProcess)<br> and (mfgProcess_id11 is hasProcess of design_id0)<br> and (hasPart of design_id0 is hasTargetPart of mfgProcess_id11)<br>(physicalResource_id16 is a PhysicalResource)<br> and (physicalResource_id16 is hasTool of grinding_id15)<br><br>If<br>( (undercut_id2 is a Undercut)<br> and (undercut_id2 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback37 isNegative true<br>feedback37 has comment "Avoid relief grooves, because they usually involve separate operations of specially ground tools.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 38 | Given<br>(design_id0 is a MechnicalDesignService)<br>(screw-machining_id1 is a Screw-machining)<br>and (screw-machining_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of screw-machining_id1)<br>(groove_id7 is a Groove)<br>and (groove_id7 is hasFeature of hasPart of design_id0)<br>(sideWall_id4 is a SideWall)<br>and (sideWall_id4 is hasSidewall of groove_id7)<br>(axis_id13 is a Axis)<br>and (axis_id13 is hasAxis of workpiece_id17)<br>(workpiece_id17 is a Workpiece)<br>and (workpiece_id17 is hasProcessInput of screw-machining_id1)<br><br>If<br>(groove_id7 isPerpendicularTo axis_id13)<br>not (hasDraft of sideWall_id4 > 0.5)<br><br>Then<br>feedback38 isNegative true<br>feedback38 has comment "Draft of sidewalls of grooves that are perpendicular to the axis of the workpiece should be 0.5 or more.". |
| 39 | Given<br>(design_id0 is a MechnicalDesignService)<br>(screw-machining_id1 is a Screw-machining)<br>and (screw-machining_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of screw-machining_id1)<br>(knurl_id5 is a Knurl)<br>and (knurl_id5 is hasTool of screw-machining_id1)<br><br>If<br>(hasWidth of knurl_id5 > hasDiameter of null)<br><br>Then<br>feedback39 isNegative true<br>feedback39 has comment "Width of a knurl should not exceed its diameter.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 40 | Given<br>(design_id0 is a MechnicalDesignService)<br>(screw-machining_id7 is a Screw-machining)<br>and (screw-machining_id7 is hasProcess of design_id0)<br>(part_id9 is a Part)<br>and (part_id9 is hasTargetPart of screw-machining_id7)<br>and (part_id9 is hasPart of design_id0)<br><br>If<br>(corner_id3 isSharp true)<br>( (corner_id3 is a Corner)<br>and (corner_id3 is hasFeature of part_id9) )<br><br>Then<br>feedback40 isNegative true<br>feedback40 has comment "Avoid sharp corners in screw-machined parts.". |
| 41 | Given<br>(design_id0 is a MechnicalDesignService)<br><br>If<br>(cutting_id5 isIrregular true)<br>(cutting_id5 isInterrupted true)<br>( (cutting_id5 is a Cutting)<br>and (cutting_id5 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of cutting_id5) )<br><br>Then<br>feedback41 isNegative true<br>feedback41 has comment "An irregular and interrupted cutting should be avoided.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 42 | Given<br>(design_id0 is a MechnicalDesignService)<br>(turning_id1 is a Turning)<br>and (turning_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of turning_id1)<br><br>If<br>(corner_id4 isSharp true)<br>( (corner_id4 is a Corner)<br>and (corner_id4 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback42 isNegative true<br>feedback42 has comment "Avoid sharp corners.". |
| 43 | Given<br>(design_id0 is a MechnicalDesignService)<br>(drilling_id1 is a Drilling)<br>and (drilling_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of drilling_id1)<br>(twistDrill_id20 is a TwistDrill)<br>and (twistDrill_id20 is hasTool of drilling_id1)<br>(twistDrill_id27 is a TwistDrill)<br>and (twistDrill_id27 is hasTool of drilling_id1)<br>(hole_id9 is a Hole)<br>and (hole_id9 is hasFeature of hasPart of design_id0)<br>(exitSurface_id3 is a ExitSurface)<br>and (exitSurface_id3 is hasExitSurface of hole_id9)<br>(axis_id16 is a Axis)<br>and (axis_id16 is hasAxis of twistDrill_id20)<br><br>If<br>(null avoid true)<br>not (exitSurface_id3 isPerpendicularTo axis_id16)<br><br>Then<br>feedback43 isNegative true<br>feedback43 has comment "The exit surface of a hole also should be perpendicular to the axis of the drill to avoid breakage problems as the drill leaves the work.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 44 | Given<br>(design_id0 is a MechnicalDesignService)<br>(drilling_id1 is a Drilling)<br>and (drilling_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of drilling_id1)<br>(blindHole_id3 is a BlindHole)<br>and (blindHole_id3 is hasFeature of hasPart of design_id0)<br>(bottomSurface_id14 is a BottomSurface)<br><br>If<br>(bottomSurface_id14 isFlat true)<br>(bottomSurface_id14 is hasBottomSurface of they(Unclassified))<br><br>Then<br>feedback44 isNegative true<br>feedback44 has comment "When blind holes are specified, they should not have flat bottoms.". |
| 45 | Given<br>(design_id0 is a MechnicalDesignService)<br>(drilling_id1 is a Drilling)<br>and (drilling_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of drilling_id1)<br><br>If<br>(hole_id4 isDeep true)<br>( (hole_id4 is a Hole)<br>and (hole_id4 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback45 isNegative true<br>feedback45 has comment "Avoid deep holes because of chip-clearance problems and the possibility of deviations from straightness.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 46 | Given<br>(design_id0 is a MechnicalDesignService)<br>(reaming_id2 is a Reaming)<br>and (reaming_id2 is hasProcess of design_id0)<br><br>If<br>(reaming_id2 hasTargetFeature hole_id4)<br>( (hole_id4 is a Hole)<br>and (hole_id4 is hasTargetFeature of reaming_id2)<br>and (hole_id4 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback46 isNegative true<br>feedback46 has comment "Avoid reamed holes .". |
| 47 | Given<br>(design_id0 is a MechnicalDesignService)<br>(drilling_id4 is a Drilling)<br>and (drilling_id4 is hasProcess of design_id0)<br>(hole_id6 is a Hole)<br>and (hole_id6 is hasTargetFeature of drilling_id4)<br>and (hole_id6 is hasFeature of hasPart of design_id0)<br><br>If<br>(hasDepth of hole_id6 > 3*hasDiameter of null)<br><br>Then<br>feedback47 isNegative true<br>feedback47 has comment "Depth of a drilled hole should not exceed 3 times its diameter.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 48 | Given<br>(design_id0 is a MechnicalDesignService)<br>(drilling_id8 is a Drilling)<br>and (drilling_id8 is hasProcess of design_id0)<br>(hole_id10 is a Hole)<br>and (hole_id10 is hasTargetFeature of drilling_id8)<br>and (hole_id10 is hasFeature of hasPart of design_id0)<br>(axis_id16 is a Axis)<br>and (axis_id16 is hasGeometry of hasFeature of hasPart of design_id0)<br>(entranceSurface_id2 is a EntranceSurface)<br>and (entranceSurface_id2 is hasEntranceSurface of hole_id10)<br><br>If<br>not (entranceSurface_id2 isPerpendicularTo axis_id16)<br><br>Then<br>feedback48 isNegative true<br>feedback48 has comment "The entrance surface of a drilled hole should be perpendicular to its axis.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 49 | Given<br>(design_id0 is a MechnicalDesignService)<br>(faceMilling_id24 is a FaceMilling)<br>and (faceMilling_id24 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of faceMilling_id24)<br>(hole_id17 is a Hole)<br>and (hole_id17 is hasFeature of hasPart of design_id0)<br>(surface_id6 is a Surface)<br>and (surface_id6 is hasGeometry of hasFeature of hasPart of design_id0)<br>(entranceSurface_id11 is a EntranceSurface)<br>and (entranceSurface_id11 is hasEntranceSurface of hole_id17)<br><br>If<br>(surface_id6 isPerpendicularTo entranceSurface_id11)<br>(surface_id6 isSmall true)<br>(surface_id6 isFlat true)<br><br>Then<br>feedback49 isNegative true<br>feedback49 has comment "When a small, flat surface is perpendicular to an entrance surface of a hole, spotfacing is more economical than face milling.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 50 | Given<br>(design_id0 is a MechnicalDesignService)<br>(milling_id1 is a Milling)<br> and (milling_id1 is hasProcess of design_id0)<br> and (hasPart of design_id0 is hasTargetPart of milling_id1)<br>(part_id10 is a Part)<br> and (part_id10 is hasPart of design_id0)<br>(externalCorner_id4 is a ExternalCorner)<br> and (externalCorner_id4 is hasFeature of part_id10)<br><br>If<br>(<br> (externalCorner_id4 isSharp true) or<br> (is internal(Unclassified) isSharp true)<br>)<br><br>Then<br>feedback50 isNegative true<br>feedback50 has comment "Internal and external corners of a part should not be sharp.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 51 | Given<br>(design_id0 is a MechnicalDesignService)<br>(milling_id1 is a Milling)<br>and (milling_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of milling_id1)<br>(cuttingTool_id18 is a CuttingTool)<br>and (cuttingTool_id18 is hasTool of milling_id1)<br>(slot_id2 is a Slot)<br>and (slot_id2 is hasFeature of hasPart of design_id0)<br>(chamfer_id8 is a Chamfer)<br>and (chamfer_id8 is hasFeature of hasPart of design_id0)<br><br>If<br>(cuttingTool_id18 isStandard true)<br>(<br>not (hasRadius of null == those(Unclassified)) or<br>not (hasWidth of slot_id2 == those(Unclassified)) or<br>not (chamfer_id8 == those(Unclassified))<br>)<br><br>Then<br>feedback51 isNegative true<br>feedback51 has comment "Slot widths, radii, and chamfers should conform to those of standard cutters.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 52 | Given<br>(design_id0 is a MechnicalDesignService)<br>(endMilling_id9 is a EndMilling)<br> and (endMilling_id9 is hasProcess of design_id0)<br>(cuttingTool_id18 is a CuttingTool)<br> and (cuttingTool_id18 is hasTool of endMilling_id9)<br>(slot_id11 is a Slot)<br> and (slot_id11 is hasTargetFeature of endMilling_id9)<br> and (slot_id11 is hasFeature of hasPart of design_id0)<br>(steel_id2 is a Steel)<br> and (steel_id2 is isMadeOf of hasPart of design_id0)<br><br>If<br>(hasDepth of slot_id11 > hasDiameter of cuttingTool_id18)<br><br>Then<br>feedback52 isNegative true<br>feedback52 has comment "For steel, the depth of end-milled slots should not be deeper than the cutter diameter.". |
| 53 | Given<br>(design_id12 is a MechnicalDesignService)<br>(milling_id8 is a Milling)<br> and (milling_id8 is hasProcess of design_id12)<br>(shoulder_id6 is a Shoulder)<br> and (shoulder_id6 is hasFeature of hasPart of design_id12)<br>(surface_id2 is a Surface)<br> and (surface_id2 is hasTargetGeometry of milling_id8)<br> and (surface_id2 is hasGeometry of hasFeature of hasPart of design_id12)<br>(cutterPath_id18 is a CutterPath)<br> and (cutterPath_id18 is hasProcessInput of milling_id8)<br><br>If<br>(surface_id2 isAdjacentTo shoulder_id6)<br>not (hasClearance of cutterPath_id18 > 0)<br><br>Then<br>feedback53 isNegative true<br>feedback53 has comment "When surfaces adjacent to a shoulder are milled, the design should have clearance for the cutter path.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 54 | Given<br>(design_id0 is a MechnicalDesignService)<br>(milling_id1 is a Milling)<br>and (milling_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of milling_id1)<br>(part_id3 is a Part)<br>and (part_id3 is hasPart of design_id0)<br>(step_id7 is a Step)<br>(flange_id12 is a Flange)<br>(shoulderSurface_id14 is a ShoulderSurface)<br><br>If<br>(step_id7 isSmall true)<br>(flange_id12 isInclined true)<br>(<br>not (shoulderSurface_id14 is hasGeometry of hasFeature of part_id3) or<br>not (hasRadius of part_id3 > 0) or<br>not (flange_id12 is hasFeature of part_id3) or<br>not (step_id7 is hasFeature of part_id3)<br>)<br><br>Then<br>feedback54 isNegative true<br>feedback54 has comment "The part should have small steps or radii or inclined flange or shoulder surfaces.". |
| 55 | Given<br>(design_id0 is a MechnicalDesignService)<br>(mechanicalSubtraction_id12 is a MechanicalSubtraction)<br>and (mechanicalSubtraction_id12 is hasProcess of design_id0)<br>(rail_id14 is a Rail)<br>and (rail_id14 is hasTargetPart of mechanicalSubtraction_id12)<br>and (rail_id14 is hasPart of design_id0)<br>(blendedRadius_id8 is a BlendedRadius)<br>and (blendedRadius_id8 is hasFeature of rail_id14)<br><br>Then<br>feedback55 isNegative true<br>feedback55 has comment "It is better not to specify a blended radius on machined rails.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 56 | Given<br>(design_id0 is a MechnicalDesignService)<br>(endMilling_id11 is a EndMilling)<br>and (endMilling_id11 is hasProcess of design_id0)<br>(cuttingTool_id21 is a CuttingTool)<br>and (cuttingTool_id21 is hasTool of endMilling_id11)<br>(slot_id13 is a Slot)<br>and (slot_id13 is hasTargetFeature of endMilling_id11)<br>and (slot_id13 is hasFeature of hasPart of design_id0)<br>(mildSteel_id2 is a MildSteel)<br>and (mildSteel_id2 is isMadeOf of hasPart of design_id0)<br><br>if<br>(hasDepth of slot_id13 > hasDiameter of cuttingTool_id21)<br><br>Then<br>feedback56 isNegative true<br>feedback56 has comment "For mild steel, the depth of end-milled slots should not exceed the diameter of the cutter.". |
| 57 | Given<br>(design_id0 is a MechnicalDesignService)<br>(planing_id1 is a Planing)<br>and (planing_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of planing_id1)<br>(cuttingTool_id15 is a CuttingTool)<br>and (cuttingTool_id15 is hasTool of planing_id1)<br>(cutterPath_id9 is a CutterPath)<br>and (cutterPath_id9 is hasCutterPath of cuttingTool_id15)<br><br>If<br>not (surface_id3 isParallelTo cutterPath_id9)<br>( (surface_id3 is a Surface)<br>and (surface_id3 is hasGeometry of hasFeature of hasPart of design_id0))<br><br>Then<br>feedback57 isNegative true<br>feedback57 has comment "Avoid surfaces that are not parallel to cutter path of the cutting tool because this would necessitate additional setups.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 58 | Given<br>(design_id0 is a MechnicalDesignService)<br>(planing_id1 is a Planing)<br>and (planing_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of planing_id1)<br>(contour_id6 is a Contour)<br>and (contour_id6 is hasFeature of hasPart of design_id0)<br>(hole_id13 is a Hole)<br>and (hole_id13 is hasFeature of hasPart of design_id0)<br>(slot_id4 is a Slot)<br>and (slot_id4 is hasFeature of hasPart of design_id0)<br><br>If<br>(hasLength of slot_id4 > 4*hasDiameter of hole_id13)<br><br>Then<br>feedback58 isNegative true<br>feedback58 has comment “Length of slots and contours should not exceed 4 times the hole diameter.”. |
| 59 | Given<br>(design_id0 is a MechnicalDesignService)<br>(threadGrinding_id18 is a ThreadGrinding)<br>and (threadGrinding_id18 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of threadGrinding_id18)<br><br>If<br>(<br>(hasFlatness of null > 0) or<br>(hasTaper of null > 0)<br>)<br><br>Then<br>feedback59 isNegative true<br>feedback59 has comment “Taper and flatness also should be avoided because they will not be removed by the thread-grinding.”. |

Table D.1 (cont')

| Id | Extracted rule |
| --- | --- |
| 60 | Given<br>(design_id0 is a MechnicalDesignService)<br>(cutting_id8 is a Cutting)<br>and (cutting_id8 is hasProcess of design_id0)<br>(section_id5 is a Section)<br>and (section_id5 is hasTargetFeature of cutting_id8)<br>and (section_id5 is hasFeature of hasPart of design_id0)<br>(surface_id1 is a Surface)<br>and (surface_id1 is hasGeometry of hasFeature of hasPart of design_id0)<br><br>If<br>(surface_id1 isAdjacentTo section_id5)<br>(cutting_id8 toBePerformed true)<br>not (surface_id1 isSquare true)<br><br>Then<br>feedback60 isNegative true<br>feedback60 has comment "Surfaces adjacent to the area to be cut should be square and flat.". |
| 61 | Given<br>(design_id0 is a MechnicalDesignService)<br>(broaching_id1 is a Broaching)<br>and (broaching_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of broaching_id1)<br>(hole_id3 is a Hole)<br>and (hole_id3 is hasFeature of hasPart of design_id0)<br><br>If<br>(hole_id3 isLong true)<br>not (hole_id3 isChambered true)<br><br>Then<br>feedback61 isNegative true<br>feedback61 has comment "Long holes should be chambered to improve accuracy as well as to reduce costs.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 62 | Given<br>(design_id0 is a MechnicalDesignService)<br>(broaching_id1 is a Broaching)<br>and (broaching_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of broaching_id1)<br><br>If<br>(<br>( (dovetail_id2 is a Dovetail)<br>and (dovetail_id2 is hasFeature of hasPart of design_id0) ) or<br>( (inverted-angleSpline_id6 is a Inverted-angleSpline)<br>and (inverted-angleSpline_id6 is hasFeature of hasPart of design_id0) )<br>)<br><br>Then<br>feedback62 isNegative true<br>feedback62 has comment "Dovetail or inverted-angle splines should be avoided.". |
| 63 | Given<br>(design_id0 is a MechnicalDesignService)<br>(broaching_id1 is a Broaching)<br>and (broaching_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of broaching_id1)<br><br>If<br>(hasTaper of spline_id3 $>$ 0)<br>( (spline_id3 is a Spline)<br>and (spline_id3 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback63 isNegative true<br>feedback63 has comment "Tapered splines should be avoided.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 64 | Given<br>(design_id0 is a MechnicalDesignService)<br>(blindHole_id3 is a BlindHole)<br>and (blindHole_id3 is hasTargetFeature of broaching_id1)<br>and (blindHole_id3 is hasFeature of hasPart of design_id0)<br><br>If<br>( (broaching_id1 is a Broaching)<br>and (broaching_id1 is hasProcess of design_id0) )<br><br>Then<br>feedback64 isNegative true<br>feedback64 has comment "Broaching blind holes should be avoided.". |
| 65 | Given<br>(design_id0 is a MechnicalDesignService)<br>(broaching_id1 is a Broaching)<br>and (broaching_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of broaching_id1)<br>(point_id12 is a Point)<br>and (point_id12 is hasPoint of stress_id11)<br>(stress_id11 is a Stress)<br>If<br>(internalCorner_id3 isSharp true)<br>( (internalCorner_id3 is a InternalCorner)<br>and (internalCorner_id3 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback65 isNegative true<br>feedback65 has comment "Sharp internal corners should be avoided to eliminate stress points and minimize tooth-edge wear.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 66 | Given<br>(design_id0 is a MechnicalDesignService)<br>(broaching_id5 is a Broaching)<br>and (broaching_id5 is hasProcess of design_id0)<br>(externalSurface_id7 is a ExternalSurface)<br>and (externalSurface_id7 is hasTargetGeometry of broaching_id5)<br>and (externalSurface_id7 is hasGeometry of hasFeature of hasPart of design_id0)<br><br>If<br>(externalSurface_id7 hasCorner corner_id2)<br>(corner_id2 isSharp true)<br>( (corner_id2 is a Corner)<br>and (corner_id2 is hasCorner of externalSurface_id7) )<br><br>Then<br>feedback66 isNegative true<br>feedback66 has comment "Sharp corners of broached outer surfaces should be avoided.". |
| 67 | Given<br>(design_id0 is a MechnicalDesignService)<br>(contourSawing_id3 is a ContourSawing)<br>and (contourSawing_id3 is hasProcess of design_id0)<br><br>If<br>(contourSawing_id3 hasTargetFeature hole_id5)<br>( (hole_id5 is a Hole)<br>and (hole_id5 is hasTargetFeature of contourSawing_id3)<br>and (hole_id5 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback67 isNegative true<br>feedback67 has comment "Contour-sawed holes should be avoided.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 68 | Given<br>(design_id0 is a MechnicalDesignService)<br>(flatGrinding_id3 is a FlatGrinding)<br>and (flatGrinding_id3 is hasProcess of design_id0)<br><br>If<br>(flatGrinding_id3 hasTargetGeometry bottomSurface_id5)<br>( (bottomSurface_id5 is a BottomSurface)<br>and (bottomSurface_id5 is hasTargetGeometry of flatGrinding_id3)<br>and (bottomSurface_id5 is hasGeometry of hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback68 isNegative true<br>feedback68 has comment "Flat-grinding bottoms should be avoided.". |
| 69 | Given<br>(design_id0 is a MechnicalDesignService)<br>(centerlessGrinding_id4 is a CenterlessGrinding)<br>and (centerlessGrinding_id4 is hasProcess of design_id0)<br>(part_id7 is a Part)<br>and (part_id7 is hasTargetPart of centerlessGrinding_id4)<br>and (part_id7 is hasPart of design_id0)<br><br>If<br>not (hasWidth of null > hasDiameter of null)<br><br>Then feedback69 isNegative true<br>feedback69 has comment "When centerless-grinding a part, its width should be greater than its diameter.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 70 | Given<br>(design_id0 is a MechnicalDesignService)<br>(rollerBurnishing_id11 is a RollerBurnishing)<br>and (rollerBurnishing_id11 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of rollerBurnishing_id11)<br>(blindHole_id7 is a BlindHole)<br>and (blindHole_id7 is hasFeature of hasPart of design_id0)<br><br>Then<br>feedback70 isNegative true<br>feedback70 has comment "Through-holes are preferable to blind holes if roller burnishing is performed.". |
| 71 | Given<br>(design_id0 is a MechnicalDesignService)<br>(heatTreating_id3 is a HeatTreating)<br>and (heatTreating_id3 is hasProcess of design_id0)<br>(gear_id5 is a Gear)<br>and (gear_id5 is hasTargetPart of heatTreating_id3)<br>and (gear_id5 is hasPart of design_id0)<br>(crossSection_id9 is a CrossSection)<br><br>If<br>(crossSection_id9 isUniform true)<br>not (crossSection_id9 is hasCrossSection of gear_id5)<br><br>Then<br>feedback71 isNegative true<br>feedback71 has comment "Heat-treated gears should have uniform cross section to minimize heat-treatment distortion.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 72 | Given<br>(design_id0 is a MechnicalDesignService)<br>(casting_id1 is a Casting)<br>and (casting_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of casting_id1)<br>(pattern_id7 is a Pattern)<br>and (pattern_id7 is hasFeature of hasPart of design_id0)<br><br>If<br>(<br>not (hasDraft of pattern_id7 $>$ 0) or<br>not (hasTaper of pattern_id7 $>$ 0)<br>)<br><br>Then<br>feedback72 isNegative true<br>feedback72 has comment "To facilitate removal, the pattern must have taper or draft.". |
| 73 | Given<br>(design_id0 is a MechnicalDesignService)<br>(casting_id1 is a Casting)<br>and (casting_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of casting_id1)<br>(section_id7 is a Section)<br>and (section_id7 is hasFeature of hasPart of design_id0)<br><br>If<br>(section_id7 isWedged true)<br>(hasTaper of section_id7 $>$ 0.25)<br><br>Then<br>feedback73 isNegative true<br>feedback73 has comment "The taper of the wedged area should not exceed 0.25.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 74 | Given<br>(design_id0 is a MechnicalDesignService)<br>(mechanicalSubtraction_id8 is a MechanicalSubtraction)<br>and (mechanicalSubtraction_id8 is hasProcess of design_id0)<br>(casting_id12 is a Casting)<br>and (casting_id12 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of casting_id12)<br><br>If<br>(mechanicalSubtraction_id8 hasTargetFeature corner_id3)<br>(corner_id3 isSharp true)<br>(mechanicalSubtraction_id8 toBePerformed true)<br>(<br>( (corner_id3 is a Corner)<br>and (corner_id3 is hasTargetFeature of mechanicalSubtraction_id8)<br>and (corner_id3 is hasFeature of hasPart of design_id0) ) or<br>( (fin_id5 is a Fin)<br>and (fin_id5 is hasFeature of hasPart of design_id0) )<br>)<br><br>Then<br>feedback74 isNegative true<br>feedback74 has comment "Avoid sharp corners and fins to be machined after casting.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 75 | Given<br>(design_id0 is a MechnicalDesignService)<br>(mechanicalSubtraction_id4 is a MechanicalSubtraction)<br>and (mechanicalSubtraction_id4 is hasProcess of design_id0)<br>(surface_id1 is a Surface)<br>and (surface_id1 is hasTargetGeometry of mechanicalSubtraction_id4)<br>and (surface_id1 is hasGeometry of hasFeature of hasPart of design_id0)<br><br>If<br>(mechanicalSubtraction_id4 toBePerformed true)<br>not (hasMachiningAllowance of surface_id1 > 0)<br><br>Then<br>feedback75 isNegative true<br>feedback75 has comment "Surfaces to be machined must have stock allowance.". |
| 76 | Given<br>(design_id0 is a MechnicalDesignService)<br>(casting_id1 is a Casting)<br>and (casting_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of casting_id1)<br>(key_id9 is a Key)<br>and (key_id9 is hasPart of design_id0)<br>(ferrous_id3 is a Ferrous)<br>and (key_id9 isMadeOf ferrous_id3)<br><br>If<br>(hasWidth of key_id9 < 2.3)<br><br>Then<br>feedback76 isNegative true<br>feedback76 has comment "For ferrous metals, width of keys should not be smaller than 2.3.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 77 | Given<br>(design_id0 is a MechnicalDesignService)<br>(casting_id1 is a Casting)<br>and (casting_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of casting_id1)<br>(key_id13 is a Key)<br>and (key_id13 is hasPart of design_id0)<br>(non-ferrousAlloy_id3 is a Non-ferrousAlloy)<br>and (key_id13 isMadeOf non-ferrousAlloy_id3)<br><br>If<br>(key_id13 isCastable true)<br>not (hasWidth of key_id13 > 1.5)<br><br>Then<br>feedback77 isNegative true<br>feedback77 has comment "For nonferrous alloys, the minimum width of a castable key is 1.5.". |
| 78 | Given<br>(design_id0 is a MechnicalDesignService)<br>(casting_id1 is a Casting)<br>and (casting_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of casting_id1)<br>(blindHole_id6 is a BlindHole)<br>and (blindHole_id6 is hasFeature of hasPart of design_id0)<br><br>If<br>(hasDepth of null > 0)<br>(hasLength of null > hasDiameter of null)<br><br>Then<br>feedback78 isNegative true<br>feedback78 has comment "However, if a blind hole is required, the length of depth should not exceed the diameter.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 79 | Given<br>(design_id0 is a MechnicalDesignService)<br>(casting_id1 is a Casting)<br>and (casting_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of casting_id1)<br>(rib_id9 is a Rib)<br>and (rib_id9 is hasFeature of hasPart of design_id0)<br>(castingWall_id19 is a CastingWall)<br>and (castingWall_id19 is hasFeature of hasPart of design_id0)<br><br>If<br>(null avoid true)<br>(hasWidth of rib_id9 > hasThickness of castingWall_id19)<br><br>Then<br>feedback79 isNegative true<br>feedback79 has comment "To avoid sinks, the width of ribs should not be greater than the thickness of the casting wall.". |
| 80 | Given<br>(design_id0 is a MechnicalDesignService)<br>(casting_id1 is a Casting)<br>and (casting_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of casting_id1)<br>(core_id13 is a Core)<br>and (core_id13 is hasTool of casting_id1)<br>(core_id3 is a Core)<br>and (core_id3 is hasTool of casting_id1)<br>(aluminum_id9 is a Aluminum)<br>and (aluminum_id9 is isMadeOf of hasPart of design_id0)<br><br>If<br>(core_id13 isSmall true)<br>not (hasDiameter of core_id3 > 3)<br><br>Then<br>feedback80 isNegative true<br>feedback80 has comment "Minimum core diameter should be 3 for aluminum, since smaller cores are prone to frequent breakage and erosion from heat buildup.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 81 | Given<br>(design_id0 is a MechnicalDesignService)<br>(part_id8 is a Part)<br>and (part_id8 is hasPart of design_id0)<br>(thermosetting-plastic_id1 is a Thermosetting-plastic)<br>and (part_id8 isMadeOf thermosetting-plastic_id1)<br><br>If<br>( (internalUndercut_id2 is a InternalUndercut)<br>and (internalUndercut_id2 is hasFeature of part_id8) )<br><br>Then<br>feedback81 isNegative true<br>feedback81 has comment "Internal relief grooves in a part should be avoided.". |
| 82 | Given<br>(design_id0 is a MechnicalDesignService)<br>(compression-molding_id6 is a Compression-molding)<br>and (compression-molding_id6 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of compression-molding_id6)<br>(blindHole_id12 is a BlindHole)<br>and (blindHole_id12 is hasFeature of hasPart of design_id0)<br>(thermosetting-plastic_id1 is a Thermosetting-plastic)<br>and (thermosetting-plastic_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(hasDepth of blindHole_id12 > 1.5*hasDiameter of null)<br><br>Then<br>feedback82 isNegative true<br>feedback82 has comment "When the compression-molding is used, blind-hole depths should not exceed 1.5 times the diameter.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 83 | Given<br>(design_id0 is a MechnicalDesignService)<br>(casting_id2 is a Casting)<br>and (casting_id2 is hasProcess of design_id0)<br>(sideCore_id14 is a SideCore)<br>and (sideCore_id14 is hasTool of casting_id2)<br>(thermosetting-plastic_id1 is a Thermosetting-plastic)<br>and (thermosetting-plastic_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(casting_id2 hasTargetFeature sideHole_id4)<br>(null have null)<br>( (sideHole_id4 is a SideHole)<br>and (sideHole_id4 is hasTargetFeature of casting_id2)<br>and (sideHole_id4 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback83 isNegative true<br>feedback83 has comment "Casting side holes should be avoided since they have activated side cores.". |
| 84 | Given<br>(design_id0 is a MechnicalDesignService)<br>(wall_id19 is a Wall)<br>and (wall_id19 is hasFeature of hasPart of design_id0)<br>(rib_id10 is a Rib)<br>and (rib_id10 is hasFeature of hasPart of design_id0)<br>(base_id6 is a Base)<br>and (base_id6 is hasBase of rib_id10)<br>(thermosetting-plastic_id1 is a Thermosetting-plastic)<br>and (thermosetting-plastic_id1 is isMadeOf of hasPart of design_id0)<br><br>If not (hasWidth of base_id6 < hasThickness of wall_id19)<br><br>Then<br>feedback84 isNegative true<br>feedback84 has comment "The width of the base of the rib should be less than the thickness of the wall.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 85 | Given<br>(design_id0 is a MechnicalDesignService)<br>(boss_id3 is a Boss)<br>and (boss_id3 is hasFeature of hasPart of design_id0)<br>(thermosetting-plastic_id1 is a Thermosetting-plastic)<br>and (thermosetting-plastic_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(hasHeight of boss_id3 > 2*hasDiameter of null)<br><br>Then<br>feedback85 isNegative true<br>feedback85 has comment "The boss height should not be more than 2 times the diameter.". |
| 86 | Given<br>(design_id0 is a MechnicalDesignService)<br>(blindHole_id6 is a BlindHole)<br>and (blindHole_id6 is hasFeature of hasPart of design_id0)<br>(thermoplastic_id1 is a Thermoplastic)<br>and (thermoplastic_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(hasDepth of blindHole_id6 > 2*hasDiameter of null)<br><br>Then<br>feedback86 isNegative true<br>feedback86 has comment "The depth of a blind hole should not exceed 2 times its diameter.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 87 | Given<br>(design_id0 is a MechnicalDesignService)<br>(blindHole_id4 is a BlindHole)<br>and (blindHole_id4 is hasFeature of hasPart of design_id0)<br>(thermoplastic_id1 is a Thermoplastic)<br>and (thermoplastic_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(hasDiameter of null < 1.5)<br>not (hasDepth of null == hasDiameter of null)<br><br>Then<br>feedback87 isNegative true<br>feedback87 has comment "For a blind hole, If the diameter is smaller than 1.5, the maximum depth should be equal to the diameter.". |
| 88 | Given<br>(design_id0 is a MechnicalDesignService)<br>(mold_id19 is a Mold)<br>and (mold_id19 is hasTool of hasProcess of design_id0)<br>(part_id16 is a Part)<br>and (part_id16 is hasPart of design_id0)<br>(partingLine_id8 is a PartingLine)<br>and (partingLine_id8 is hasFeature of part_id16)<br>(rib_id2 is a Rib)<br>and (rib_id2 is hasFeature of part_id16)<br>(thermoplastic_id1 is a Thermoplastic)<br>and (part_id16 isMadeOf thermoplastic_id1)<br><br>If<br>(null have removal_id13(Unclassified))<br>not (rib_id2 isPerpendicularTo partingLine_id8)<br><br>Then<br>feedback88 isNegative true<br>feedback88 has comment "Ribs should be perpendicular to the parting line to have removal of the part from the mold.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 89 | Given<br>(design_id0 is a MechnicalDesignService)<br>(insert_id5 is a Insert)<br>and (insert_id5 is hasTool of hasProcess of design_id0)<br>(thermoplastic_id1 is a Thermoplastic)<br>and (thermoplastic_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>not (hasDepth of insert_id5 > 2*hasDiameter of null)<br><br>Then<br>feedback89 isNegative true<br>feedback89 has comment "The depth of insertion should exceed 2 times its diameter.". |
| 90 | Given<br>(design_id0 is a MechnicalDesignService)<br>(insert_id9 is a Insert)<br>and (insert_id9 is hasTool of hasProcess of design_id0)<br>(boss_id28 is a Boss)<br>and (boss_id28 is hasFeature of hasPart of design_id0)<br>(thermoplastic_id1 is a Thermoplastic)<br>and (thermoplastic_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(hasOutsideDiameter of insert_id9 < 6)<br>not (hasOutsideDiameter of insert_id9 == 0.5*hasOutsideDiameter of boss_id28)<br><br>Then<br>feedback90 isNegative true<br>feedback90 has comment 'If the external diameter of an insert is smaller than 6, it should be equal to 0.5 times the external diameter of the boss.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 91 | Given<br>(design_id0 is a MechnicalDesignService)<br>(insert_id7 is a Insert)<br>and (insert_id7 is hasTool of hasProcess of design_id0)<br>(insert_id25 is a Insert)<br>and (insert_id25 is hasTool of hasProcess of design_id0)<br>(boss_id14 is a Boss)<br>and (boss_id14 is hasFeature of hasPart of design_id0)<br>(thermoplastic_id1 is a Thermoplastic)<br>and (thermoplastic_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(hasDiameter of insert_id7 $<$ 6.25)<br>not (hasDiameter of boss_id14 $>$ 2*hasDiameter of insert_id25)<br><br>Then<br>feedback91 isNegative true<br>feedback91 has comment "If the diameter of an insert is 6.25 or less, the boss diameter should exceed 2 times the diameter of the insert.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 92 | Given<br>(design_id0 is a MechnicalDesignService)<br>(structural-foam-molding_id2 is a Structural-foam-molding)<br>and (structural-foam-molding_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of structural-foam-molding_id2)<br>(core_id9 is a Core)<br>and (core_id9 is hasTool of structural-foam-molding_id2)<br>(section_id6 is a Section)<br>and (section_id6 is hasFeature of hasPart of design_id0)<br>(boss_id3 is a Boss)<br>and (boss_id3 is hasFeature of hasPart of design_id0)<br>(non-metal_id1 is a Non-metal)<br>and (non-metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(section_id6 isThick true)<br><br>Then<br>feedback92 isNegative true<br>feedback92 has comment "Bosses and thick sections should be cored.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 93 | Given<br>(design_id0 is a MechnicalDesignService)<br>(structural-foam-molding_id2 is a Structural-foam-molding)<br>and (structural-foam-molding_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of structural-foam-molding_id2)<br>(part_id16 is a Part)<br>and (part_id16 is hasPart of design_id0)<br>(connectingRib_id5 is a ConnectingRib)<br>and (connectingRib_id5 is hasFeature of part_id16)<br>(wall_id17 is a Wall)<br>and (wall_id17 is hasWall of part_id16)<br>(non-metal_id1 is a Non-metal)<br>and (part_id16 isMadeOf non-metal_id1)<br><br>If<br>not (hasThickness of connectingRib_id5 == hasThickness of wall_id17)<br><br>Then<br>feedback93 isNegative true<br>feedback93 has comment "Thickness of connecting ribs should be equal to the thickness of the part walls.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 94 | Given<br>(design_id0 is a MechnicalDesignService)<br>(molding_id2 is a Molding)<br>and (molding_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of molding_id2)<br>(externalCorner_id5 is a ExternalCorner)<br>and (externalCorner_id5 is hasFeature of hasPart of design_id0)<br>(polymer_id1 is a Polymer)<br>and (polymer_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(<br>not (externalCorner_id5 isRadiused true) or<br>not (is internal(Unclassified) isRadiused true)<br>)<br><br>Then<br>feedback94 isNegative true<br>feedback94 has comment "Internal or external corners should be well radiused.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 95 | Given<br>(design_id0 is a MechnicalDesignService)<br>(rotationalMolding_id8 is a RotationalMolding)<br>and (rotationalMolding_id8 is hasProcess of design_id0)<br>(part_id10 is a Part)<br>and (part_id10 is hasTargetPart of rotationalMolding_id8)<br>and (part_id10 is hasPart of design_id0)<br>(polymer_id1 is a Polymer)<br>and (part_id10 isMadeOf polymer_id1)<br><br>If<br>(projection_id4 isNarrow true)<br>(<br>( (projection_id4 is a Projection)<br>and (projection_id4 is hasFeature of part_id10) ) or<br>( (recess_id6 is a Recess)<br>and (recess_id6 is hasFeature of part_id10) )<br>)<br><br>Then<br>feedback95 isNegative true<br>feedback95 has comment "Avoid narrow projections or recesses in rotomolded parts.". |
| 96 | Given<br>(design_id0 is a MechnicalDesignService)<br>(composite_id1 is a Composite)<br>and (composite_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>not (hasDraft of null > 0.5)<br><br>Then<br>feedback96 isNegative true<br>feedback96 has comment "The minimum draft angle is 0.5.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 97 | Given<br>(design_id0 is a MechnicalDesignService)<br>(pressMolding_id3 is a PressMolding)<br> and (pressMolding_id3 is hasProcess of design_id0)<br> and (hasPart of design_id0 is hasTargetPart of pressMolding_id3)<br>(composite_id1 is a Composite)<br> and (composite_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>( (undercut_id7 is a Undercut)<br> and (undercut_id7 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback97 isNegative true<br>feedback97 has comment "For press molding, relief grooves should be avoided.". |
| 98 | Given<br>(design_id0 is a MechnicalDesignService)<br>(pressing_id2 is a Pressing)<br> and (pressing_id2 is hasProcess of design_id0)<br> and (hasPart of design_id0 is hasTargetPart of pressing_id2)<br>(hole_id17 is a Hole)<br> and (hole_id17 is hasFeature of hasPart of design_id0)<br>(shoulder_id7 is a Shoulder)<br> and (shoulder_id7 is hasFeature of hasPart of design_id0)<br>(composite_id1 is a Composite)<br> and (composite_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(hasWidth of shoulder_id7 > hasDiameter of hole_id17)<br><br>Then<br>feedback98 isNegative true<br>feedback98 has comment "The width of the shoulder should not be greater than the diameter of the hole.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 99 | Given<br>(design_id0 is a MechnicalDesignService)<br>(surface_id10 is a Surface)<br>and (surface_id10 is hasGeometry of hasFeature of hasPart of design_id0)<br>(composite_id1 is a Composite)<br>and (composite_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(surface_id10 isFlat true)<br><br>Then<br>feedback99 isNegative true<br>feedback99 has comment "Curved or rounded surfaces are preferred to flat surfaces.". |
| 100 | Given<br>(design_id0 is a MechnicalDesignService)<br>(forming_id12 is a Forming)<br>and (forming_id12 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of forming_id12)<br>(point_id21 is a Point)<br>and (point_id21 is hasGeometry of hasFeature of hasPart of design_id0)<br>(polymer_id27 is a Polymer)<br>and (polymer_id27 is isMadeOf of hasPart of design_id0)<br><br>If<br>(notch_id14(Unclassified) is point_id21)<br>(internalCorner_id2 isSharp true)<br>( (internalCorner_id2 is a InternalCorner)<br>and (internalCorner_id2 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback100 isNegative true<br>feedback100 has comment "Sharp internal corners also should be avoided because they can form a notch that can be an easy breaking point for the more rigid plastics.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 101 | Given<br>(design_id0 is a MechnicalDesignService)<br>(extruding_id2 is a Extruding)<br>and (extruding_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of extruding_id2)<br>(wall_id10 is a Wall)<br>and (wall_id10 is hasFeature of hasPart of design_id0)<br>(corner_id3 is a Corner)<br>and (corner_id3 is hasFeature of hasPart of design_id0)<br>(polymer_id1 is a Polymer)<br>and (polymer_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>not (hasRadius of corner_id3 > 1.5*hasThickness of wall_id10)<br><br>Then<br>feedback101 isNegative true<br>feedback101 has comment "Corner radii should exceed 1.5 times the wall thickness.". |
| 102 | Given<br>(design_id0 is a MechnicalDesignService)<br>(straightVacuumForming_id3 is a StraightVacuumForming)<br>and (straightVacuumForming_id3 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of straightVacuumForming_id3)<br>(femaleMold_id9 is a FemaleMold)<br>and (femaleMold_id9 is hasTool of straightVacuumForming_id3)<br>(polymer_id1 is a Polymer)<br>and (polymer_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(hasDepth of null > 0)<br>(hasWidth of null > 0)<br>(hasDepth of null/hasWidth of null > 0.5)<br><br>Then<br>feedback102 isNegative true<br>feedback102 has comment "For straight vacuum forming with a female mold, the depth-to-width ratio should not exceed 0.5.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 103 | Given<br>(design_id0 is a MechnicalDesignService)<br>(drapeForming_id3 is a DrapeForming)<br>and (drapeForming_id3 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of drapeForming_id3)<br>(maleMold_id8 is a MaleMold)<br>and (maleMold_id8 is hasTool of drapeForming_id3)<br>(polymer_id1 is a Polymer)<br>and (polymer_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(hasDepth of null > 0)<br>(hasWidth of null > 0)<br>(hasDepth of null/hasWidth of null > 1)<br><br>Then<br>feedback103 isNegative true<br>feedback103 has comment "For drape forming with a male mold, the depth-to-width ratio should not exceed 1.". |
| 104 | Given<br>(design_id0 is a MechnicalDesignService)<br>(polymer_id1 is a Polymer)<br>and (polymer_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(undercut_id3 isDeep true)<br>( (undercut_id3 is a Undercut)<br>and (undercut_id3 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback104 isNegative true<br>feedback104 has comment "Deep relief grooves should be avoided.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 105 | Given<br>(design_id0 is a MechnicalDesignService)<br>(molding_id30 is a Molding)<br>and (molding_id30 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of molding_id30)<br>(forming_id14 is a Forming)<br>and (forming_id14 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of forming_id14)<br>(pin_id13 is a Pin) and (pin_id13 is hasTool of molding_id30) (blindHole_id8 is a BlindHole)<br>and (blindHole_id8 is hasFeature of hasPart of design_id0)<br>(end_id21 is a End)<br>and (end_id21 is hasFeature of hasPart of design_id0)<br>(rubber_id1 is a Rubber)<br>and (rubber_id1 is isMadeOf of hasPart of design_id0)<br><br>Then<br>feedback105 isNegative true<br>feedback105 has comment "Through-holes are preferable to blind holes because the pins forming them can be anchored at both ends and are therefore less susceptible to deflection during molding.". |
| 106 | Given<br>(design_id0 is a MechnicalDesignService)<br>(rubber_id1 is a Rubber)<br>and (rubber_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>( (undercut_id2 is a Undercut)<br>and (undercut_id2 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback106 isNegative true<br>feedback106 has comment "Relief grooves should be avoided.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 107 | Given<br>(design_id0 is a MechnicalDesignService)<br>(rubber_id1 is a Rubber)<br>and (rubber_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>not (hasRadius of null > 1.5*hasOutsideDiameter of null)<br><br>Then<br>feedback107 isNegative true<br>feedback107 has comment "The minimum radius should exceed 1.5 times the external diameter.". |
| 108 | Given<br>(design_id0 is a MechnicalDesignService)<br>(rubber_id1 is a Rubber)<br>and (rubber_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>( (featheredge_id3 is a Featheredge)<br>and (featheredge_id3 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback108 isNegative true<br>feedback108 has comment "Avoid featheredges.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 109 | Given<br>(design_id0 is a MechnicalDesignService)<br>(mechanicalSubtraction_id5 is a MechanicalSubtraction)<br>and (mechanicalSubtraction_id5 is hasProcess of design_id0)<br>(part_id3 is a Part)<br>and (part_id3 is hasTargetPart of mechanicalSubtraction_id5)<br>and (part_id3 is hasPart of design_id0)<br>(ceramic_id1 is a Ceramic)<br>and (part_id3 isMadeOf ceramic_id1)<br><br>If<br>not (hasOutsideRadius of null > 1.5)<br><br>Then<br>feedback109 isNegative true<br>feedback109 has comment "When parts are machined, external radii should be greater than 1.5.". |
| 110 | Given<br>(design_id0 is a MechnicalDesignService)<br>(dry-pressing_id5 is a Dry-pressing)<br>and (dry-pressing_id5 is hasProcess of design_id0)<br>(part_id23 is a Part)<br>and (part_id23 is hasPart of design_id0)<br>(part_id7 is a Part)<br>and (part_id7 is hasTargetPart of dry-pressing_id5)<br>and (part_id7 is hasPart of design_id0)<br>(outsideEdge_id9 is a OutsideEdge)<br>and (outsideEdge_id9 is hasFeature of part_id23)<br>(ceramic_id1 is a Ceramic)<br>and (part_id23 isMadeOf ceramic_id1)<br><br>If<br>not (outsideEdge_id9 isBeveled true)<br><br>Then<br>feedback110 isNegative true<br>feedback110 has comment "For dry-pressed parts, external edges should be beveled similar to that employed with powder-metal parts.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 111 | Given<br>(design_id0 is a MechnicalDesignService)<br>(section_id4 is a Section)<br>and (section_id4 is hasFeature of hasPart of design_id0)<br>(ceramic_id1 is a Ceramic)<br>and (ceramic_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(hasThickness of section_id4 > 25)<br><br>Then<br>feedback111 isNegative true<br>feedback111 has comment "Thickness of sections should not exceed 25.". |
| 112 | Given<br>(design_id0 is a MechnicalDesignService)<br>(part_id9 is a Part)<br>and (part_id9 is hasPart of design_id0)<br>(ceramic_id8 is a Ceramic)<br>and (part_id9 isMadeOf ceramic_id8)<br><br>If<br>( (undercut_id1 is a Undercut)<br>and (undercut_id1 is hasFeature of part_id9) )<br><br>Then<br>feedback112 isNegative true<br>feedback112 has comment "Relief grooves should be avoided in ceramic parts.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 113 | Given<br>(design_id0 is a MechnicalDesignService)<br>(extruding_id3 is a Extruding)<br>and (extruding_id3 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of extruding_id3)<br>(wall_id7 is a Wall)<br>and (wall_id7 is hasFeature of hasPart of design_id0)<br>(ceramic_id1 is a Ceramic)<br>and (ceramic_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>not (hasThickness of wall_id7 > 0.4)<br><br>Then<br>feedback113 isNegative true<br>feedback113 has comment "For extrusion, the minimum wall thickness should be 0.4.". |
| 114 | Given<br>(design_id0 is a MechnicalDesignService)<br>(mechanicalSubtraction_id4 is a MechanicalSubtraction)<br>and (mechanicalSubtraction_id4 is hasProcess of design_id0)<br>(hole_id6 is a Hole)<br>and (hole_id6 is hasTargetFeature of mechanicalSubtraction_id4)<br>and (hole_id6 is hasFeature of hasPart of design_id0)<br>(hole_id13 is a Hole)<br>and (hole_id13 is hasFeature of hasPart of design_id0)<br>(ceramic_id1 is a Ceramic)<br>and (ceramic_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(hole_id13 isSmall true)<br>not (hasDiameter of hole_id6 > 1.5)<br><br>Then<br>feedback114 isNegative true<br>feedback114 has comment "Diameter of machined holes should exceed 1.5 , although smaller holes can be produced.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 115 | Given<br>(design_id0 is a MechnicalDesignService)<br>(tapping_id3 is a Tapping)<br>and (tapping_id3 is hasProcess of design_id0)<br>(hole_id5 is a Hole)<br>and (hole_id5 is hasTargetFeature of tapping_id3)<br>and (hole_id5 is hasFeature of hasPart of design_id0)<br>(ceramic_id1 is a Ceramic)<br>and (ceramic_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>not (hole_id5 isCountersunk true)<br><br>Then<br>feedback115 isNegative true<br>feedback115 has comment "All tapped holes should be countersunk.". |
| 116 | Given<br>(design_id0 is a MechnicalDesignService)<br>(fin_id4 is a Fin)<br>and (fin_id4 is hasFeature of hasPart of design_id0)<br>(rib_id2 is a Rib)<br>and (rib_id2 is hasFeature of hasPart of design_id0)<br>(ceramic_id1 is a Ceramic)<br>and (ceramic_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(<br>not (fin_id4 isRounded true) or<br>not (rib_id2 isRounded true)<br>)<br><br>Then<br>feedback116 isNegative true<br>feedback116 has comment "Ribs and fins should be well rounded.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 117 | Given<br>(design_id0 is a MechnicalDesignService)<br>(thermalWelding_id1 is a ThermalWelding)<br>and (thermalWelding_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of thermalWelding_id1)<br><br>If<br>(jointArea_id5 isDeep true)<br>(jointArea_id5 isNarrow true)<br>( (jointArea_id5 is a JointArea)<br>and (jointArea_id5 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback117 isNegative true<br>feedback117 has comment "Deep, narrow joint areas should be avoided.". |
| 118 | Given<br>(design_id0 is a MechnicalDesignService)<br>(thermalWelding_id1 is a ThermalWelding)<br>and (thermalWelding_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of thermalWelding_id1)<br>(aluminum_id3 is a Aluminum)<br>and (aluminum_id3 is isMadeOf of hasPart of design_id0)<br>(stack_id7 is a Stack)<br>and (stack_id7 is hasProcessInput of thermalWelding_id1)<br><br>If<br>not (hasHeight of stack_id7 < 8)<br><br>Then<br>feedback118 isNegative true<br>feedback118 has comment "With aluminum, the maximum stack height is 8.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 119 | Given<br>(design_id0 is a MechnicalDesignService)<br>(flashWelding_id14 is a FlashWelding)<br>and (flashWelding_id14 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of flashWelding_id14)<br>(tube_id2 is a Tube)<br>and (tube_id2 is hasPart of design_id0)<br>(wall_id3 is a Wall)<br>and (wall_id3 is hasWall of tube_id2)<br><br>If<br>(hasThickness of wall_id3 > 5)<br>not (hasThickness of wall_id3 isBeveled true)<br><br>Then<br>feedback119 isNegative true<br>feedback119 has comment "If tube wall thickn ess exceeds 5, it should be beveled before flash welding.". |
| 120 | Given<br>(design_id0 is a MechnicalDesignService)<br>(thermalWelding_id1 is a ThermalWelding)<br>and (thermalWelding_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of thermalWelding_id1)<br><br>If<br>( (angleWeld_id2 is a AngleWeld)<br>and (angleWeld_id2 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback120 isNegative true<br>feedback120 has comment "Angle welds should be avoided because the upsetting force is not applied so squarely to the joint.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 121 | Given<br>(design_id0 is a MechnicalDesignService)<br>(thermalWelding_id1 is a ThermalWelding)<br> and (thermalWelding_id1 is hasProcess of design_id0)<br> and (hasPart of design_id0 is hasTargetPart of thermalWelding_id1)<br>(tube_id3 is a Tube)<br> and (tube_id3 is hasPart of design_id0)<br>(section_id5 is a Section)<br> and (section_id5 is hasFeature of tube_id3)<br><br>If<br>(hasJointAngle of null < 150)<br><br>Then<br>feedback121 isNegative true<br>feedback121 has comment "For tubing and sections, the joint angle should not be smaller than 150.". |
| 122 | Given<br>(design_id0 is a MechnicalDesignService)<br>(finishing_id1 is a Finishing)<br> and (finishing_id1 is hasProcess of design_id0)<br> and (hasPart of design_id0 is hasTargetPart of finishing_id1)<br>(internalCorner_id3 is a InternalCorner)<br> and (internalCorner_id3 is hasFeature of hasPart of design_id0)<br><br>If<br>not (internalCorner_id3 isRounded true)<br><br>Then<br>feedback122 isNegative true<br>feedback122 has comment "All internal corners should be rounded.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 123 | Given<br>(design_id0 is a MechnicalDesignService)<br>(finishing_id1 is a Finishing)<br>and (finishing_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of finishing_id1)<br>(heat-treatingStress_id12 is a Heat-treatingStress)<br><br>If<br>(internalCorner_id4 isSharp true)<br>( (internalCorner_id4 is a InternalCorner)<br>and (internalCorner_id4 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback123 isNegative true<br>feedback123 has comment "Avoid sharp internal corners, which concentrate heat-treating stresses.". |
| 124 | Given<br>(paintMasking_id7 is a PaintMasking)<br><br>If<br>(paintMasking_id7 is hasProcess of design_id2)<br>(hasPart of design_id2 is hasTargetPart of paintMasking_id7)<br>( (design_id2 is a PartDesign) )<br><br>Then<br>feedback124 isNegative true<br>feedback124 has comment "Avoid part designs that have paint masking.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 125 | Given<br>(design_id0 is a MechnicalDesignService)<br>(painting_id8 is a Painting)<br>and (painting_id8 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of painting_id8)<br>(film_id20 is a Film)<br>and (film_id20 is hasFeature of hasPart of design_id0)<br>(section_id15 is a Section)<br>and (section_id15 is hasFeature of hasPart of design_id0)<br><br>If<br>(edge_id3 isSharp true)<br>(film_id20 isThin true)<br>(<br>( (edge_id3 is a Edge)<br>and (edge_id3 is hasFeature of hasPart of design_id0) ) or<br>( (point_id5 is a Point)<br>and (point_id5 is hasGeometry of hasFeature of hasPart of design_id0) )<br>)<br><br>Then<br>feedback125 isNegative true<br>feedback125 has comment "Avoid sharp edges and points, because paint tends to creep away from these areas and leave too thin film.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 126 | Given<br>(design_id0 is a MechnicalDesignService)<br>(finishing_id1 is a Finishing)<br>and (finishing_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of finishing_id1)<br>(corner_id13 is a Corner)<br>and (corner_id13 is hasFeature of hasPart of design_id0)<br><br>If<br>( (blindArea_id3 is a BlindArea)<br>and (blindArea_id3 is hasFeature of hasPart of design_id0) )<br><br>then<br>feedback126 isNegative true<br>feedback126 has comment "Avoid blind areas, i.e., surfaces that are behind corners or protrusions.". |
| 127 | Given<br>(design_id0 is a MechnicalDesignService)<br>(finishing_id1 is a Finishing)<br>and (finishing_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of finishing_id1)<br><br>If<br>(recess_id3 isDeep true)<br>(recess_id3 isNarrow true)<br>( (recess_id3 is a Recess)<br>and (recess_id3 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback127 isNegative true<br>feedback127 has comment "Deep recesses, however, should be avoided, particularly if they are narrow.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 128 | Given<br>(design_id0 is a MechnicalDesignService)<br>(curtainCoating_id3 is a CurtainCoating)<br>and (curtainCoating_id3 is hasProcess of design_id0)<br>(part_id5 is a Part)<br>and (part_id5 is hasTargetPart of curtainCoating_id3)<br>and (part_id5 is hasPart of design_id0)<br>(depression_id9 is a Depression)<br><br>If (depression_id9 is hasFeature of part_id5)<br><br>Then<br>feedback128 isNegative true<br>feedback128 has comment "Curtain-coated parts should not have depressions.". |
| 129 | Given<br>(design_id0 is a MechnicalDesignService)<br>(coating_id1 is a Coating)<br>and (coating_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of coating_id1)<br><br>If<br>(edge_id3 isSharp true)<br>(<br>( (edge_id3 is a Edge)<br>and (edge_id3 is hasFeature of hasPart of design_id0) ) or<br>( (verticalCorner_id5 is a VerticalCorner)<br>and (verticalCorner_id5 is hasFeature of hasPart of design_id0) )<br>)<br><br>Then<br>feedback129 isNegative true<br>feedback129 has comment "Sharp edges and vertical corners also should be avoided.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 130 | Given<br>(design_id0 is a MechnicalDesignService)<br>(electrostaticPowder-spraying_id3 is a ElectrostaticPowder-spraying)<br>and (electrostaticPowder-spraying_id3 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of electrostaticPowder-spraying_id3)<br><br>If<br>(edge_id12 isSharp true)<br>(<br>( (edge_id12 is a Edge)<br>and (edge_id12 is hasFeature of hasPart of design_id0) ) or<br>( (recess_id14 is a Recess)<br>and (recess_id14 is hasFeature of hasPart of design_id0) )<br>)<br><br>Then<br>feedback130 isNegative true<br>feedback130 has comment "If an electrostatic powder-spraying is used, sharp edges and recesses should be avoided.". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 131 | Given<br>(design_id0 is a MechnicalDesignService)<br>(marking_id1 is a Marking)<br>and (marking_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of marking_id1)<br><br>If<br>(recess_id7 isDeep true)<br>(recess_id7 isNarrow true)<br>(<br>( (overhang_id2 is a Overhang)<br>and (overhang_id2 is hasFeature of hasPart of design_id0) ) or<br>( (recess_id7 is a Recess)<br>and (recess_id7 is hasFeature of hasPart of design_id0) )<br>)<br><br>Then<br>feedback131 isNegative true<br>feedback131 has comment "Overhangs and deep and narrow recesses should be avoided.". |
| 132 | Given<br>(design_id0 is a MechnicalDesignService)<br>(marking_id1 is a Marking)<br>and (marking_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of marking_id1)<br>(workpiece_id9 is a Workpiece)<br>and (workpiece_id9 is hasProcessInput of marking_id1)<br><br>If<br>not (hasStroke of null > 3*hasThickness of workpiece_id9)<br><br>Then<br>feedback132 isNegative true<br>feedback132 has comment "The stroke should exceed 3 times the stock thickness .". |

Table D.1 (cont')

| Id | Extracted rule |
|---|---|
| 133 | Given<br>(design_id0 is a MechnicalDesignService)<br>(finishing_id1 is a Finishing)<br>and (finishing_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of finishing_id1)<br><br>If<br>(internalCorner_id4 isSharp true)<br>( (internalCorner_id4 is a InternalCorner)<br>and (internalCorner_id4 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback133 isNegative true<br>feedback133 has comment "Avoid sharp internal corners.". |

## APPENDIX E: THE DETAILS OF AMBIGUITY RESOLUTION RESULT

Table E.1 shows the full list of the additional correct rules extracted after adopting the ambiguity resolution method (Chapter 3).

Table E.1: The list of the additional correct rules extracted after adopting the ambiguity resolution method (Chapter 3)

| Id | Extracted rule |
|---|---|
| 2 | Given<br>(design_id0 is a MechnicalDesignService)<br>(slot_id5 is a Slot)<br>and (slot_id5 is hasFeature of hasPart of design_id0)<br><br>If<br>not (hasWidth of slot_id5 > hasDepth of slot_id5)<br><br>Then<br>feedback2 isNegative true<br>feedback2 has comment “The width of a slot should exceed its depth.”. |
| 3 | Given<br>(design_id0 is a MechnicalDesignService)<br><br>If<br>(corner_id2 isSharp true)<br>(edge_id5 isSharp true)<br>(<br>( (edge_id5 is a Edge)<br>and (edge_id5 is hasFeature of hasPart of design_id0) ) or<br>( (corner_id2 is a Corner)<br>and (corner_id2 is hasFeature of hasPart of design_id0) )<br>)<br><br>Then<br>feedback3 isNegative true<br>feedback3 has comment “Sharp corners and edges should be avoided.”. |

Table E.1 (cont')

| Id | Extracted rule |
|---|---|
| 14 | Given<br>(design_id0 is a MechnicalDesignService)<br>(stamping_id2 is a Stamping)<br>and (stamping_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of stamping_id2)<br>(punch_id12 is a Punch)<br>and (punch_id12 is hasTool of stamping_id2)<br>(hole_id17 is a Hole)<br>and (hole_id17 is hasTargetFeature of punch_id12)<br>and (hole_id17 is hasFeature of hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(hole_id17 isSquare true)<br><br>Then<br>feedback14 isNegative true<br>feedback14 has comment "Punches for round holes are more economical than those for square holes.". |
| 28 | Given<br>(design_id0 is a MechnicalDesignService)<br>(extruding_id2 is a Extruding)<br>and (extruding_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of extruding_id2)<br>(flange_id6 is a Flange)<br>and (flange_id6 is hasFeature of hasPart of design_id0)<br>(sideWall_id15 is a SideWall)<br>and (sideWall_id15 is hasFeature of hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>not (hasThickness of flange_id6 > hasThickness of sideWall_id15)<br><br>Then<br>feedback28 isNegative true<br>feedback28 has comment "Thickness of the flange should be greater than that of the sidewall.". |

Table E.1 (cont')

| Id | Extracted rule |
|---|---|
| 39 | Given<br>(design_id0 is a MechnicalDesignService)<br>(screw-machining_id1 is a Screw-machining)<br>and (screw-machining_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of screw-machining_id1)<br>(knurl_id5 is a Knurl)<br>and (knurl_id5 is hasTool of screw-machining_id1)<br><br>If<br>(hasWidth of knurl_id5 > hasDiameter of knurl_id5)<br><br>Then<br>feedback39 isNegative true<br>feedback39 has comment "Width of a knurl should not exceed its diameter.". |
| 44 | Given<br>(design_id0 is a MechnicalDesignService)<br>(drilling_id1 is a Drilling)<br>and (drilling_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of drilling_id1)<br>(blindHole_id3 is a BlindHole)<br>and (blindHole_id3 is hasFeature of hasPart of design_id0)<br>(bottomSurface_id14 is a BottomSurface)<br><br>If<br>(bottomSurface_id14 isFlat true)<br>(bottomSurface_id14 is hasBottomSurface of blindHole_id3)<br><br>Then<br>feedback44 isNegative true<br>feedback44 has comment "When blind holes are specified, they should not have flat bottoms.". |

Table E.1 (cont')

| Id | Extracted rule |
|---|---|
| 47 | Given<br>(design_id0 is a MechnicalDesignService)<br>(drilling_id4 is a Drilling)<br>and (drilling_id4 is hasProcess of design_id0)<br>(hole_id6 is a Hole)<br>and (hole_id6 is hasTargetFeature of drilling_id4)<br>and (hole_id6 is hasFeature of hasPart of design_id0)<br><br>If<br>(hasDepth of hole_id6 > 3*hasDiameter of hole_id6)<br><br>Then<br>feedback47 isNegative true<br>feedback47 has comment "Depth of a drilled hole should not exceed 3 times its diameter.". |
| 48 | Given<br>(design_id0 is a MechnicalDesignService)<br>(drilling_id8 is a Drilling)<br>and (drilling_id8 is hasProcess of design_id0)<br>(hole_id10 is a Hole)<br>and (hole_id10 is hasTargetFeature of drilling_id8)<br>and (hole_id10 is hasFeature of hasPart of design_id0)<br>(axis_id16 is a Axis)<br>and (axis_id16 is hasAxis of hole_id10)<br>(entranceSurface_id2 is a EntranceSurface)<br>and (entranceSurface_id2 is hasEntranceSurface of hole_id10)<br><br>If<br>not (entranceSurface_id2 isPerpendicularTo axis_id16)<br><br>Then<br>feedback48 isNegative true<br>feedback48 has comment "The entrance surface of a drilled hole should be perpendicular to its axis.". |

Table E.1 (cont')

| Id | Extracted rule |
|---|---|
| 50 | Given<br>(design_id0 is a MechnicalDesignService)<br>(milling_id1 is a Milling)<br>and (milling_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of milling_id1)<br>(part_id12 is a Part)<br>and (part_id12 is hasPart of design_id0)<br>(internalCorner_id2 is a InternalCorner)<br>and (internalCorner_id2 is hasFeature of part_id12)<br>(externalCorner_id6 is a ExternalCorner)<br>and (externalCorner_id6 is hasFeature of part_id12)<br><br>If<br>(<br>(internalCorner_id2 isSharp true) or<br>(externalCorner_id6 isSharp true)<br>)<br><br>Then<br>feedback50 isNegative true<br>feedback50 has comment "Internal and external corners of a part should not be sharp.". |

Table E.1 (cont')

| Id | Extracted rule |
|---|---|
| 51 | Given<br>(design_id0 is a MechnicalDesignService)<br>(milling_id1 is a Milling)<br>and (milling_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of milling_id1)<br>(cuttingTool_id20 is a CuttingTool)<br>and (cuttingTool_id20 is hasTool of milling_id1)<br>(chamfer_id15 is a Chamfer)<br>and (chamfer_id15 is hasChamfer of cuttingTool_id20)<br>(slot_id2 is a Slot)<br>and (slot_id2 is hasFeature of hasPart of design_id0)<br>(chamfer_id8 is a Chamfer)<br>and (chamfer_id8 is hasChamfer of slot_id2)<br><br>If<br>(cuttingTool_id20 isStandard true)<br>(<br>not (hasRadius of slot_id2 == hasRadius of cuttingTool_id20) or<br>not (hasWidth of slot_id2 == hasWidth of cuttingTool_id20) or<br>not (chamfer_id8 == chamfer_id15)<br>)<br><br>Then<br>feedback51 isNegative true<br>feedback51 has comment "Slot widths, radii, and chamfers should conform to those of standard cutters.". |

Table E.1 (cont')

| Id | Extracted rule |
|---|---|
| 69 | Given<br>(design_id0 is a MechnicalDesignService)<br>(centerlessGrinding_id4 is a CenterlessGrinding)<br>and (centerlessGrinding_id4 is hasProcess of design_id0)<br>(part_id7 is a Part)<br>and (part_id7 is hasTargetPart of centerlessGrinding_id4)<br>and (part_id7 is hasPart of design_id0)<br><br>If<br>not (hasWidth of part_id7 > hasDiameter of part_id7)<br><br>Then<br>feedback69 isNegative true<br>feedback69 has comment "When centerless-grinding a part, its width should be greater than its diameter.". |
| 86 | Given<br>(design_id0 is a MechnicalDesignService)<br>(blindHole_id6 is a BlindHole)<br>and (blindHole_id6 is hasFeature of hasPart of design_id0)<br>(thermoplastic_id1 is a Thermoplastic)<br>and (thermoplastic_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(hasDepth of blindHole_id6 > 2*hasDiameter of blindHole_id6)<br><br>Then<br>feedback86 isNegative true<br>feedback86 has comment "The depth of a blind hole should not exceed 2 times its diameter.". |

Table E.1 (cont')

| Id | Extracted rule |
|---|---|
| 89 | Given<br>(design_id0 is a MechnicalDesignService)<br>(insert_id5 is a Insert)<br> and (insert_id5 is hasTool of hasProcess of design_id0)<br>(thermoplastic_id1 is a Thermoplastic)<br> and (thermoplastic_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>not (hasDepth of insert_id5 > 2*hasDiameter of insert_id5)<br><br>Then<br>feedback89 isNegative true<br>feedback89 has comment "The depth of insertion should exceed 2 times its diameter.". |
| 94 | Given<br>(design_id0 is a MechnicalDesignService)<br>(molding_id2 is a Molding)<br> and (molding_id2 is hasProcess of design_id0)<br> and (hasPart of design_id0 is hasTargetPart of molding_id2)<br>(internalCorner_id3 is a InternalCorner)<br> and (internalCorner_id3 is hasFeature of hasPart of design_id0)<br>(externalCorner_id7 is a ExternalCorner)<br> and (externalCorner_id7 is hasFeature of hasPart of design_id0)<br>(polymer_id1 is a Polymer)<br> and (polymer_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(<br> not (internalCorner_id3 isRadiused true) or<br> not (externalCorner_id7 isRadiused true)<br>)<br><br>Then<br>feedback94 isNegative true<br>feedback94 has comment "Internal or external corners should be well radiused.". |

Table E.1 (cont')

| Id | Extracted rule |
|---|---|
| 95 | Given<br>(design_id0 is a MechnicalDesignService)<br>(rotationalMolding_id9 is a RotationalMolding)<br>and (rotationalMolding_id9 is hasProcess of design_id0)<br>(part_id11 is a Part)<br>and (part_id11 is hasTargetPart of rotationalMolding_id9)<br>and (part_id11 is hasPart of design_id0)<br>(polymer_id1 is a Polymer)<br>and (part_id11 isMadeOf polymer_id1)<br><br>If<br>(projection_id4 isNarrow true)<br>(recess_id7 isNarrow true)<br>(<br>( (projection_id4 is a Projection)<br>and (projection_id4 is hasFeature of part_id11) ) or<br>( (recess_id7 is a Recess)<br>and (recess_id7 is hasFeature of part_id11) )<br>)<br><br>Then<br>feedback95 isNegative true<br>feedback95 has comment "Avoid narrow projections or recesses in rotomolded parts.". |

Table E.1 (cont')

| Id | Extracted rule |
|---|---|
| 119 | Given<br>(design_id0 is a MechnicalDesignService)<br>(flashWelding_id14 is a FlashWelding)<br>and (flashWelding_id14 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of flashWelding_id14)<br>(tube_id2 is a Tube)<br>and (tube_id2 is hasPart of design_id0)<br>(wall_id3 is a Wall)<br>and (wall_id3 is hasWall of tube_id2)<br><br>If<br>(hasThickness of wall_id3 > 5)<br>not (wall_id3 isBeveled true)<br><br>Then<br>feedback119 isNegative true<br>feedback119 has comment "If tube wall thickness exceeds 5, it should be beveled before flash welding.". |
| 130 | Given<br>(design_id0 is a MechnicalDesignService)<br>(electrostaticPowder-spraying_id3 is a ElectrostaticPowder-spraying)<br>and (electrostaticPowder-spraying_id3 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of electrostaticPowder-spraying_id3)<br><br>If<br>(edge_id12 isSharp true)<br>(recess_id15 isSharp true)<br>(<br>( (edge_id12 is a Edge)<br>and (edge_id12 is hasFeature of hasPart of design_id0) ) or<br>( (recess_id15 is a Recess)<br>and (recess_id15 is hasFeature of hasPart of design_id0) )<br>)<br><br>Then<br>feedback130 isNegative true<br>feedback130 has comment "If an electrostatic powder-spraying is used, sharp edges and recesses should be avoided.". |

# APPENDIX F: THE DETAILS OF FEEDBACK GENERATION RESULT

Table F.1 shows the full list of the additional correct rules extracted after manually modifying the input sentences based on the feedback generation method (Chapter 4).

Table F.1: The list of additional correct rules extracted after manually modifying the input sentences based on the feedback generation method (Chapter 4)

<table>
<tr><th>Id</th><th colspan="2">Results</th></tr>
<tr><td rowspan="3">1</td><td>Feedback</td><td>-Input sentence and the part needs revision:<br>Sharp corners should be avoided <u>because they interfere with smooth metal flow</u> .<br><br>-Suggestion:<br>The clause possibly has unnecessary information. Consider enclosing the clause with parenthesis so that it is not processed.</td></tr>
<tr><td>Modified sentence</td><td>Sharp Corners should be avoided <u>(because they interfere with metal flow)</u>.</td></tr>
<tr><td>Extracted rule</td><td>Given<br>(design_id0 is a MechnicalDesignService)<br><br>If<br>(corner_id2 isSharp true)<br>( (corner_id2 is a Corner)<br>and (corner_id2 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback134 isNegative true<br>feedback134 has comment “Sharp Corners should be avoided (because they interfere with metal flow).”.</td></tr>
</table>

Table F.1 (cont')

| Id | Results | |
|---|---|---|
| 4 | Feedback | -Input sentence and the part needs revision:<br>It is better not to have sharp corners on a formed part .<br><br>-Suggestion:<br>The "it" may be a pleonastic pronoun. Consider revising it to have an explicit subject. e.g. <[corners/part] [should have / should be have-ed]>. |
| | Modified sentence | A formed part should not have sharp corners. |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(forming_id2 is a Forming)<br>and (forming_id2 is hasProcess of design_id0)<br>(part_id4 is a Part)<br>and (part_id4 is hasTargetPart of forming_id2)<br>and (part_id4 is hasPart of design_id0)<br>(corner_id9 is a Corner)<br><br>If<br>(corner_id9 isSharp true)<br>(corner_id9 is hasCorner of part_id4)<br><br>Then<br>feedback135 isNegative true<br>feedback135 has comment "A formed part should not have sharp corners.". |

Table F.1 (cont')

| Id | Results | |
|---|---|---|
| | Feedback | -Input sentence and the part needs revision:<br>If a blind hole is required, its length of depth should not exceed its diameter .<br><br>-Suggestion 1:<br>The attribute <length> has owner <depth>,which seems semantically invalid as it is also a numeric value. Remove the redundancy of the expression if there is, or revise the sentence/update domain ontology.<br><br>-Suggestion 2:<br>The owner of the attribute <depth> is not found. Consider revising the sentence or update domain ontology to specify the owner of the attribute. e.g. the attribute may be owned by other concepts such as... "<depth> of the [part/blind hole]" |
| | Modified sentence | If a blind hole is required, its depth should not exceed its diameter. |
| 5 | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(blindHole_id3 is a BlindHole)<br>and (blindHole_id3 is hasFeature of hasPart of design_id0)<br><br>If<br>(hasDepth of blindHole_id3 > hasDiameter of blindHole_id3)<br><br>Then<br>feedback136 isNegative true<br>feedback136 has comment "If a blind hole is required, its depth should not exceed its diameter .". |

Table F.1 (cont')

<table>
<tr><th>Id</th><th colspan="2">Results</th></tr>
<tr><td rowspan="3">6</td><td>Feedback</td><td>-Input sentence and the part needs revision:<br>If a sharp internal corner is necessary, the included angle should be always more than 90 .<br><br>-Suggestion:<br>The owner of the attribute <included angle> is not found. Consider revising the sentence or update domain ontology to specify the owner of the attribute. e.g. the attribute may be owned by other concepts such as... "<included angle> of the [part/internal corner]"</td></tr>
<tr><td>Modified sentence</td><td>If a sharp internal corner is necessary, its inclined angle should be more than 90.</td></tr>
<tr><td>Extracted rule</td><td>Given<br>(design_id0 is a MechnicalDesignService)<br>(extruding_id2 is a Extruding)<br> and (extruding_id2 is hasProcess of design_id0)<br> and (hasPart of design_id0 is hasTargetPart of extruding_id2)<br>(internalCorner_id6 is a InternalCorner)<br> and (internalCorner_id6 is hasFeature of hasPart of design_id0)<br>(metal_id1 is a Metal)<br> and (metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(internalCorner_id6 isSharp true)<br>not (hasIncludedAngle of internalCorner_id6 > 90)<br><br>Then<br>feedback137 isNegative true<br>feedback137 has comment "If a sharp internal corner is necessary, its included angle should be always more than 90 .".</td></tr>
</table>

Table F.1 (cont')

| Id | | Results |
|---|---|---|
| 7 | Feedback | -Input sentence and the part needs revision:<br>Avoid sharp edges of parts <u>because they interfere with the smooth flow of material through the die</u>.<br><br>-Suggestion:<br>The clause possibly has unnecessary information. |
| | Modified sentence | Avoid sharp edges of parts <u>(because they interfere with the smooth flow of material through the die)</u>. |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(extruding_id2 is a Extruding)<br>and (extruding_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of extruding_id2)<br>(part_id8 is a Part)<br>and (part_id8 is hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (part_id8 isMadeOf metal_id1)<br><br>If<br>(part_id8 hasEdge edge_id5)<br>(edge_id5 isSharp true)<br>( (edge_id5 is a Edge)<br>and (edge_id5 is hasEdge of part_id8) )<br><br>Then<br>feedback138 isNegative true<br>feedback138 has comment "Avoid sharp edges of parts (because they interfere with the smooth flow of material through the die).". |

Table F.1 (cont')

| Id | Results | |
|---|---|---|
| 10 | Feedback | -Input sentence and the part needs revision:<br>Symmetrical cross sections are preferable to nonsymmetrical cross sections to avoid unbalanced stresses and warpage.<br><br>-Suggestion:<br>The phrase led by <auxiliary to> likely has unnecessary information. Consider enclosing the phrase with parenthesis so that it is not processed. |
| | Modified sentence | Symmetrical cross sections are preferable to nonsymmetrical cross sections (to avoid unbalanced stresses and warpage). |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(extruding_id2 is a Extruding)<br>and (extruding_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of extruding_id2)<br>(crossSection_id11 is a CrossSection)<br>and (crossSection_id11 is hasFeature of hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(crossSection_id11 isNonsymmetrical true)<br><br>Then<br>feedback139 isNegative true<br>feedback139 has comment "Symmetrical cross sections are preferable to nonsymmetrical cross sections (to avoid unbalanced stresses and warpage).". |

Table F.1 (cont')

| Id | Results | |
|---|---|---|
| 15 | Feedback | -Input sentence and the part needs revision:<br>The internal bend angle should be 90 .<br><br>-Suggestion:<br>The owner of the attribute <internal bend angle> is not found. Consider revising the sentence or update domain ontology to specify the owner of the attribute. e.g. the attribute may be owned by other concepts such as... "<internal bend angle> of the [part/feature]" |
| | Modified sentence | The inside bend angle of the part should be 90 . |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(stamping_id2 is a Stamping)<br>and (stamping_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of stamping_id2)<br>(part_id10 is a Part)<br>and (part_id10 is hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (part_id10 isMadeOf metal_id1)<br><br>If<br>not (hasInsideBendAngle of part_id10 == 90)<br><br>Then<br>feedback140 isNegative true<br>feedback140 has comment "The internal bend angle of the part should be 90.". |

Table F.1 (cont')

| Id | Results | |
|---|---|---|
| 19 | Feedback | -Input sentence and the part needs revision:<br>Avoid reverse-formed designs , because they have additional operations and can cause considerable thinning of stock.<br><br>-Suggestion:<br>The clause possibly has unnecessary information. Consider enclosing the clause with parenthesis so that it is not processed. |
| | Modified sentence | Avoid reverse-formed designs, (because they have additional operations and can cause considerable thinning of stock). |
| | Extracted rule | Given<br>(spinning_id2 is a Spinning)<br>and (spinning_id2 is hasProcess of design_id7)<br>and (hasPart of design_id7 is hasTargetPart of spinning_id2)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id7)<br><br>If<br>(design_id7 isReverse-formed true)<br>( (design_id7 is a MechnicalDesignService) )<br><br>Then<br>feedback142 isNegative true<br>feedback142 has comment "Avoid reverse-formed designs, (because they have additional operations and can cause considerable thinning of stock).". |

Table F.1 (cont')

<table>
<tr><th>Id</th><th colspan="2">Results</th></tr>
<tr><td rowspan="3">20</td><td>Feedback</td><td>-Input sentence and the part needs revision:<br>Hollow upsets should be avoided <u>because they have more die maintenance and there is a possibility of cracks forming around the edges of the recesses</u>.<br><br>-Suggestion:<br>The clause possibly has unnecessary information. Consider enclosing the clause with parenthesis so that it is not processed.</td></tr>
<tr><td>Modified sentence</td><td>Hollow upsets should be avoided <u>(because they have more die maintenance and there is a possibility of cracks forming around the edges of the recesses)</u>.</td></tr>
<tr><td>Extracted rule</td><td>Given<br>(design_id0 is a MechnicalDesignService)<br>(forming_id2 is a Forming)<br>and (forming_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of forming_id2)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(upset_id4 isHollow true)<br>( (upset_id4 is a Upset)<br>and (upset_id4 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback143 isNegative true<br>feedback143 has comment "Hollow upsets should be avoided (because they have more die maintenance and there is a possibility of cracks forming around the edges of the recesses)".</td></tr>
</table>

Table F.1 (cont')

| Id | Results | |
|---|---|---|
| 21 | Feedback | -Input sentence and the part needs revision:<br>In flat-headed parts, the slot must have a curved bottom to eliminate contact between the slotting punch and the dies.<br><br>-Suggestion:<br>The phrase led by <auxiliary to> likely has unnecessary information. Consider enclosing the phrase with parenthesis so that it is not processed. |
| | Modified sentence | In flat-headed parts, the slot must have a curved bottom (to eliminate contact between the slotting punch and the dies). |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(forming_id2 is a Forming)<br>and (forming_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of forming_id2)<br>(part_id7 is a Part)<br>and (part_id7 is hasPart of design_id0)<br>(slot_id10 is a Slot)<br>and (slot_id10 is hasFeature of part_id7)<br>(bottomSurface_id15 is a BottomSurface)<br>(metal_id1 is a Metal)<br>and (part_id7 isMadeOf metal_id1)<br><br>If<br>(part_id7 isFlat-headed true)<br>(bottomSurface_id15 isCurved true)<br>not (bottomSurface_id15 is hasBottomSurface of slot_id10)<br><br>Then<br>feedback144 isNegative true<br>feedback144 has comment "In flat-headed parts, the slot must have a curved bottom (to eliminate contact between the slotting punch and the dies).". |

Table F.1 (cont')

| Id | Results | |
|---|---|---|
| 22 | Feedback | -Input sentence and the part needs revision:<br>Impact-extruded parts should be symmetrical to avoid lateral movement of tooling punches and unequal wall thickness.<br><br>-Suggestion:<br>The phrase led by <auxiliary to> likely has unnecessary information. Consider enclosing the phrase with parenthesis so that it is not processed. |
| | Modified sentence | Impact-extruded parts should be symmetrical (to avoid lateral movement of tooling punches and unequal wall thickness). |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(impactExtruding_id4 is a ImpactExtruding)<br>and (impactExtruding_id4 is hasProcess of design_id0)<br>(part_id6 is a Part)<br>and (part_id6 is hasTargetPart of impactExtruding_id4)<br>and (part_id6 is hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (part_id6 isMadeOf metal_id1)<br><br>If<br>not (part_id6 isSymmetrical true)<br><br>Then<br>feedback145 isNegative true<br>feedback145 has comment "Impact-extruded parts should be symmetrical (to avoid lateral movement of tooling punches and unequal wall thickness)". |

Table F.1 (cont')

| Id | Results | |
|---|---|---|
| 35 | Feedback | -Input sentence and the part needs revision:<br>Internal corners should be rounded to ensure an even deposit of metal.<br><br>-Suggestion:<br>The phrase led by <auxiliary to> likely has unnecessary information. Consider enclosing the phrase with parenthesis so that it is not processed. |
| | Modified sentence | Inside corners should be rounded (to ensure an even deposit of metal). |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(electroforming_id2 is a Electroforming)<br>and (electroforming_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of electroforming_id2)<br>(internalCorner_id3 is a InternalCorner)<br>and (internalCorner_id3 is hasFeature of hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>not (internalCorner_id3 isRounded true)<br><br>Then<br>feedback146 isNegative true<br>feedback146 has comment "Inside corners should be rounded (to ensure an even deposit of metal).". |

Table F.1 (cont')

| Id | Results | |
|---|---|---|
| 36 | Feedback | -Input sentence and the part needs revision:<br>Relief grooves, negative tapers, and reentrant angles especially should be avoided, because they necessitate a flexible or dissolvable mandrel that is less accurate and more costly.<br><br>-Suggestion 1:<br>The clause possibly has unnecessary information. Consider enclosing the clause with parenthesis so that it is not processed.<br><br>-Suggestion 2:<br>The owner of the attribute <negative taper/reentrant angle> is not found. Consider revising the sentence or update domain ontology to specify the owner of the attribute. e.g. the attribute may be owned by other concepts such as... "<negative taper/reentrant angle> of the [part/feature]" |
| | Modified sentence | Undercuts, negative tapers, and reentrant angles of features especially should be avoided, (because they necessitate a flexible or dissolvable mandrel that is less accurate and more costly). |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(electroforming_id2 is a Electroforming)<br>and (electroforming_id2 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of electroforming_id2)<br>(manufacturingFeature_id16 is a ManufacturingFeature)<br>and (manufacturingFeature_id16 is hasFeature of hasPart of design_id0)<br>(metal_id1 is a Metal)<br>and (metal_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>( ( (undercut_id3 is a Undercut)<br>and (undercut_id3 is hasFeature of hasPart of design_id0) ) or<br>(hasReverseTaper of manufacturingFeature_id16 > 0) or<br>(hasReentrantAngle of manufacturingFeature_id16 > 0) )<br><br>Then<br>feedback147 isNegative true<br>feedback147 has comment "Undercuts, negative tapers, ...". |

Table F.1 (cont')

<table>
<tr><th>Id</th><th colspan="2">Results</th></tr>
<tr><td rowspan="3">37</td><td>Feedback</td><td>-Input sentence and the part needs revision:<br>Avoid relief grooves, <u>because they usually involve separate operations of specially ground tools</u>.<br><br>-Suggestion:<br>The clause possibly has unnecessary information. Consider enclosing the clause with parenthesis so that it is not processed.</td></tr>
<tr><td>Modified sentence</td><td>Avoid undercuts, <u>(because they usually involve separate operations of specially ground tools)</u>.</td></tr>
<tr><td>Extracted rule</td><td>Given<br>(design_id0 is a MechnicalDesignService)<br>(mechanicalSubtraction_id1 is a MechanicalSubtraction)<br>and (mechanicalSubtraction_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of mechanical-Subtraction_id1)<br><br>If<br>( (undercut_id3 is a Undercut)<br>and (undercut_id3 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback148 isNegative true<br>feedback148 has comment "Avoid undercuts, (because they usually involve separate operations of specially ground tools).".</td></tr>
</table>

Table F.1 (cont')

| Id | | Results |
|---|---|---|
| 43 | Feedback | -Input sentence and the part needs revision:<br>The exit surface of a hole also should be perpendicular to the axis of the drill to avoid breakage problems as the drill leaves the work.<br><br>-Suggestion 1:<br>The clause possibly has unnecessary information. Consider enclosing the clause with parenthesis so that it is not processed.<br><br>-Suggestion 2:<br>The phrase led by <auxiliary to> likely has unnecessary information. Consider enclosing the phrase with parenthesis so that it is not processed. |
| | Modified sentence | The exit surface of a hole also should be perpendicular to the axis of the drill to avoid breakage problems as the drill leaves the work. |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(drilling_id1 is a Drilling)<br>and (drilling_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of drilling_id1)<br>(twistDrill_id20 is a TwistDrill)<br>and (twistDrill_id20 is hasTool of drilling_id1)<br>(hole_id9 is a Hole)<br>and (hole_id9 is hasFeature of hasPart of design_id0)<br>(axis_id16 is a Axis)<br>and (axis_id16 is hasAxis of twistDrill_id20)<br>(exitSurface_id3 is a ExitSurface)<br>and (exitSurface_id3 is hasExitSurface of hole_id9)<br><br>If<br>not (exitSurface_id3 isPerpendicularTo axis_id16)<br><br>Then<br>feedback149 isNegative true<br>feedback149 has comment "The exit surface of a hole also should be perpendicular to the axis of the drill (to avoid breakage problems as the drill leaves the work).". |

Table F.1 (cont')

| Id | | Results |
|---|---|---|
| 55 | Feedback | -Input sentence and the part needs revision:<br>It is better not to specify a blended radius on machined rails .<br><br>-Suggestion:<br>The "it" may be a pleonastic pronoun. Consider revising it to have an explicit subject. e.g. <[radius/rails] [should specify / should be specify-ed]>. |
| | Modified sentence | A machined rails should not have a blended radius. |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(mechanicalSubtraction_id2 is a MechanicalSubtraction)<br>and (mechanicalSubtraction_id2 is hasProcess of design_id0)<br>(rail_id4 is a Rail)<br>and (rail_id4 is hasTargetPart of mechanicalSubtraction_id2)<br>and (rail_id4 is hasPart of design_id0)<br>(blendedRadius_id9 is a BlendedRadius)<br><br>If<br>(blendedRadius_id9 is hasFeature of rail_id4)<br><br>Then<br>feedback150 isNegative true<br>feedback150 has comment "A machined rails should not have a blended radius.". |

Table F.1 (cont')

| Id | Results | |
|---|---|---|
| 59 | Feedback | -Input sentence and the part needs revision:<br>Taper also should be avoided because they will not be removed by the thread-grinding.<br><br>-Suggestion 1:<br>The clause possibly has unnecessary information. Consider enclosing the clause with parenthesis so that it is not processed.<br><br>-Suggestion 2:<br>The owner of the attribute <Taper> is not found. Consider revising the sentence or update domain ontology to specify the owner of the attribute. e.g. the attribute may be owned by other concepts such as... "<Taper> of the [part/feature]" |
| | Modified sentence | Taper of the features also should be avoided because they will not be removed by the thread-grinding. |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(mechanicalSubtraction_id1 is a MechanicalSubtraction)<br>and (mechanicalSubtraction_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of mechanical-Subtraction_id1)<br>(manufacturingFeature_id5 is a ManufacturingFeature)<br>and (manufacturingFeature_id5 is hasFeature of hasPart of design_id0)<br><br>If<br>(hasTaper of manufacturingFeature_id5 > 0)<br><br>Then<br>feedback151 isNegative true<br>feedback151 has comment "Taper of the features also should be avoided (because they will not be removed by the thread-grinding).". |

Table F.1 (cont')

| Id | Results | |
|---|---|---|
| 65 | Feedback | -Input sentence and the part needs revision:<br>Sharp internal corners should be avoided to eliminate stress points and minimize tooth-edge wear.<br><br>-Suggestion:<br>The phrase led by <auxiliary to> likely has unnecessary information. Consider enclosing the phrase with parenthesis so that it is not processed. |
| | Modified sentence | Sharp internal corners should be avoided (to eliminate stress points and minimize tooth-edge wear). |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(broaching_id1 is a Broaching)<br>and (broaching_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of broaching_id1)<br><br>If<br>(internalCorner_id3 isSharp true)<br>( (internalCorner_id3 is a InternalCorner)<br>and (internalCorner_id3 is hasFeature of hasPart of design_id0)<br>)<br><br>Then<br>feedback152 isNegative true<br>feedback152 has comment "Sharp internal corners should be avoided (to eliminate stress points and minimize tooth-edge wear).". |

Table F.1 (cont')

| Id | Results | |
|---|---|---|
| 78 | Feedback | -Input sentence and the part needs revision:<br>However, if a blind hole is required, the length of depth should not exceed the diameter.<br><br>-Suggestion 1:<br>The attribute <length> has owner <depth>,which seems semantically invalid as it is also a numeric value. Remove the redundancy of the expression if there is, or revise the sentence/update domain ontology.<br><br>-Suggestion 2:<br>The owner of the attribute <diameter> is not found. Consider revising the sentence or update domain ontology to specify the owner of the attribute. e.g. the attribute may be owned by other concepts such as... "<diameter> of the [part/blind hole]" |
| | Modified sentence | If a blind hole is required, its depth should not exceed its diameter. |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(casting_id1 is a Casting)<br>and (casting_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of casting_id1)<br>(blindHole_id4 is a BlindHole)<br>and (blindHole_id4 is hasFeature of hasPart of design_id0)<br><br>If<br>(hasDepth of blindHole_id4 > hasDiameter of blindHole_id4)<br><br>Then<br>feedback154 isNegative true<br>feedback154 has comment "If a blind hole is required, its depth should not exceed its diameter.". |

Table F.1 (cont')

| Id | Results | |
|---|---|---|
| 79 | Feedback | -Input sentence and the part needs revision:<br>To avoid sinks, the width of ribs should not be greater than the thickness of the casting wall .<br><br>-Suggestion:<br>The phrase led by <auxiliary to> likely has unnecessary information. Consider enclosing the phrase with parenthesis so that it is not processed. |
| | Modified sentence | (To avoid sinks, )the width of ribs should not be greater than the thickness of the casting wall. |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(casting_id1 is a Casting)<br>and (casting_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of casting_id1)<br>(rib_id5 is a Rib)<br>and (rib_id5 is hasFeature of hasPart of design_id0)<br>(castingWall_id15 is a CastingWall)<br>and (castingWall_id15 is hasFeature of hasPart of design_id0)<br><br>If<br>(hasWidth of rib_id5 > hasThickness of castingWall_id15)<br><br>Then<br>feedback155 isNegative true<br>feedback155 has comment "the width of ribs should not be greater than the thickness of the casting wall.". |

Table F.1 (cont')

| Id | | Results |
|---|---|---|
| 80 | Feedback | -Input sentence and the part needs revision:<br>Minimum core diameter should be 3 for aluminum, <u>since smaller cores are prone to frequent breakage and erosion from heat buildup</u>.<br><br>-Suggestion:<br>The clause possibly has unnecessary information. Consider enclosing the clause with parenthesis so that it is not processed. |
| | Modified sentence | Minimum core diameter should be 3 for aluminum, <u>(since smaller cores are prone to frequent breakage and erosion from heat buildup)</u>. |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(casting_id1 is a Casting)<br>and (casting_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of casting_id1)<br>(core_id3 is a Core)<br>and (core_id3 is hasTool of casting_id1)<br>(aluminum_id9 is a Aluminum)<br>and (aluminum_id9 is isMadeOf of hasPart of design_id0)<br><br>If<br>not (hasDiameter of core_id3 > 3)<br><br>Then<br>feedback156 isNegative true<br>feedback156 has comment "Minimum core diameter should be 3 for aluminum, (since smaller cores are prone to frequent breakage and erosion from heat buildup).". |

Table F.1 (cont')

| Id | | Results |
|---|---|---|
| 82 | Feedback | -Input sentence and the part needs revision:<br>When the compression-molding is used, blind-hole depths should not exceed 1.5 times the diameter.<br><br>-Suggestion:<br>The owner of the attribute <diameter> is not found.Consider revising the sentence or update domain ontology to specify the owner of the attribute. e.g. the attribute may be owned by other concepts such as... "<diameter> of the [part/blind-hole]" |
| | Modified sentence | When the compression-molding is used, blind-hole depths should not exceed its diameter. |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(compression-molding_id6 is a Compression-molding)<br>and (compression-molding_id6 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of compression-molding_id6)<br>(blindHole_id12 is a BlindHole)<br>and (blindHole_id12 is hasFeature of hasPart of design_id0)<br>(thermosetting-plastic_id1 is a Thermosetting-plastic)<br>and (thermosetting-plastic_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(hasDepth of blindHole_id12 > hasDiameter of blindHole_id12)<br><br>Then<br>feedback157 isNegative true<br>feedback157 has comment "When the compression-molding is used, blind-hole depths should not exceed its diameter.". |

Table F.1 (cont')

<table>
<tr><th>Id</th><th colspan="2">Results</th></tr>
<tr><td rowspan="3">83</td><td>Feedback</td><td>-Input sentence and the part needs revision:<br>Casting side holes should be avoided <u>since they have activated side cores</u>.<br><br>-Suggestion:<br>The clause possibly has unnecessary information. Consider enclosing the clause with parenthesis so that it is not processed.</td></tr>
<tr><td>Modified sentence</td><td>Casting side holes should be avoided <u>(since they have activated side cores)</u>.</td></tr>
<tr><td>Extracted rule</td><td>Given<br>(design_id0 is a MechnicalDesignService)<br>(casting_id2 is a Casting)<br>and (casting_id2 is hasProcess of design_id0)<br>(thermosetting-plastic_id1 is a Thermosetting-plastic)<br>and (thermosetting-plastic_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(casting_id2 hasTargetFeature sideHole_id4)<br>( (sideHole_id4 is a SideHole)<br>and (sideHole_id4 is hasTargetFeature of casting_id2)<br>and (sideHole_id4 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback158 isNegative true<br>feedback158 has comment "Casting side holes should be avoided (since they have activated side cores)".</td></tr>
</table>

Table F.1 (cont')

| Id | Results | |
|---|---|---|
| 85 | Feedback | -Input sentence and the part needs revision:<br>The boss height should not be more than 2 times the diameter .<br><br>-Suggestion:<br>The owner of the attribute <diameter> is not found. Consider revising the sentence or update domain ontology to specify the owner of the attribute. e.g. the attribute may be owned by other concepts such as... "<diameter> of the [part/boss]" |
| | Modified sentence | The boss height should not be more than 2 times its diameter. |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(boss_id3 is a Boss)<br>and (boss_id3 is hasFeature of hasPart of design_id0)<br>(thermosetting-plastic_id1 is a Thermosetting-plastic)<br>and (thermosetting-plastic_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(hasHeight of boss_id3 > 2*hasDiameter of boss_id3)<br><br>Then<br>feedback159 isNegative true<br>feedback159 has comment "The boss height should not be more than 2 times its diameter.". |

Table F.1 (cont')

| Id | Results | |
|---|---|---|
| 87 | Feedback | -Input sentence and the part needs revision:<br>For a blind hole, If the diameter is smaller than 1.5, the maximum depth should be equal to the diameter.<br><br>-Suggestion:<br>The owner of the attribute <diameter> is not found. Consider revising the sentence or update domain ontology to specify the owner of the attribute. e.g. the attribute may be owned by other concepts such as... "<diameter> of the [part/blind hole]" |
| | Modified sentence | For a blind hole, If its diameter is smaller than 1.5, its maximum depth should be equal to its diameter. |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(blindHole_id4 is a BlindHole)<br>and (blindHole_id4 is hasFeature of hasPart of design_id0)<br>(thermoplastic_id1 is a Thermoplastic)<br>and (thermoplastic_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(hasDiameter of blindHole_id4 < 1.5)<br>not (hasDepth of blindHole_id4 == hasDiameter of blind-Hole_id4)<br><br>Then<br>feedback160 isNegative true<br>feedback160 has comment "For a blind hole, If its diameter is smaller than 1.5, its maximum depth should be equal to its diameter.". |

Table F.1 (cont')

| Id | | Results |
|---|---|---|
| 88 | Feedback | -Input sentence and the part needs revision:<br>Ribs should be perpendicular to the parting line to have removal of the part from the mold .<br><br>-Suggestion:<br>The phrase led by <auxiliary to> likely has unnecessary information. Consider enclosing the phrase with parenthesis so that it is not processed. |
| | Modified sentence | Ribs should be perpendicular to the parting line (to permit removal of the part from the mold). |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(partingLine_id8 is a PartingLine)<br>and (partingLine_id8 is hasFeature of hasPart of design_id0)<br>(rib_id2 is a Rib)<br>and (rib_id2 is hasFeature of hasPart of design_id0)<br>(thermoplastic_id1 is a Thermoplastic)<br>and (thermoplastic_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>not (rib_id2 isPerpendicularTo partingLine_id8)<br><br>Then<br>feedback161 isNegative true<br>feedback161 has comment "Ribs should be perpendicular to the parting line (to permit removal of the part from the mold).". |

Table F.1 (cont')

| Id | Results | |
|---|---|---|
| 96 | Feedback | -Input sentence and the part needs revision:<br>The minimum draft angle is 0.5 .<br><br>-Suggestion:<br>The owner of the attribute <draft angle> is not found. Consider revising the sentence or update domain ontology to specify the owner of the attribute. e.g. the attribute may be owned by other concepts such as... "<draft angle> of the [part]" |
| | Modified sentence | The minimum draft angle of the features is 0.5. |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(manufacturingFeature_id9 is a ManufacturingFeature)<br>and (manufacturingFeature_id9 is hasFeature of hasPart of design_id0)<br>(composite_id1 is a Composite)<br>and (composite_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>not (hasDraft of manufacturingFeature_id9 > 0.5)<br><br>Then<br>feedback162 isNegative true<br>feedback162 has comment "The minimum draft angle of the features is 0.5.". |

Table F.1 (cont')

<table>
<tr><th>Id</th><th colspan="2">Results</th></tr>
<tr><td rowspan="3">100</td><td>Feedback</td><td>-Input sentence and the part needs revision:<br>Sharp internal corners also should be avoided ulbecause they can form a notch that can be an easy breaking point for the more rigid plastics.<br><br>-Suggestion:<br>The clause possibly has unnecessary information. Consider enclosing the clause with parenthesis so that it is not processed.</td></tr>
<tr><td>Modified sentence</td><td>Sharp inside corners also should be avoided <u>(because they can form a notch that can be an easy breaking point for the more rigid plastics)</u>.</td></tr>
<tr><td>Extracted rule</td><td>Given<br>(design_id0 is a MechnicalDesignService)<br>(extruding_id2 is a Extruding)<br> and (extruding_id2 is hasProcess of design_id0)<br> and (hasPart of design_id0 is hasTargetPart of extruding_id2)<br>(polymer_id1 is a Polymer)<br> and (polymer_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>(internalCorner_id4 isSharp true)<br>( (internalCorner_id4 is a InternalCorner)<br> and (internalCorner_id4 is hasFeature of hasPart of design_id0)<br>)<br><br>Then<br>feedback163 isNegative true<br>feedback163 has comment "Sharp inside corners also should be avoided (because they can form a notch that can be an easy breaking point for the more rigid plastics).".</td></tr>
</table>

Table F.1 (cont')

| Id | Results | |
|---|---|---|
| 102 | Feedback | -Input sentence and the part needs revision:<br>For straight vacuum forming with a female mold, the depth-to-width ratio should not exceed 0.5 .<br><br>-Suggestion:<br>The owner of the attribute <depth/width> is not found. Consider revising the sentence or update domain ontology to specify the owner of the attribute. e.g. the attribute may be owned by other concepts such as... "<depth/width> of the [part/feature]" |
| | Modified sentence | For straight vacuum forming with a female mold, the depth-to-width ratio of the part should not exceed 0.5. |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(straightVacuumForming_id3 is a StraightVacuumForming)<br>and (straightVacuumForming_id3 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of straightVacuumForming_id3)<br>(femaleMold_id9 is a FemaleMold)<br>and (femaleMold_id9 is hasTool of straightVacuumForming_id3)<br>(part_id22 is a Part)<br>and (part_id22 is hasPart of design_id0)<br>(polymer_id1 is a Polymer)<br>and (part_id22 isMadeOf polymer_id1)<br><br>If<br>(hasDepth of part_id22 > 0)<br>(hasWidth of part_id22 > 0)<br>(hasDepth of part_id22/hasWidth of part_id22 > 0.5)<br><br>Then<br>feedback164 isNegative true<br>feedback164 has comment "For straight vacuum forming with a female mold, the depth-to-width ratio of the part should not exceed 0.5.". |

Table F.1 (cont')

| Id | Results | |
|---|---|---|
| 103 | Feedback | -Input sentence and the part needs revision:<br>For drape forming with a male mold, the depth-to-width ratio should not exceed 1 .<br><br>-Suggestion:<br>The owner of the attribute <depth/width> is not found. Consider revising the sentence or update domain ontology to specify the owner of the attribute. e.g. the attribute may be owned by other concepts such as... "<depth/width> of the [part/feature]" |
| | Modified sentence | For drape forming with a male mold, the depth-to-width ratio of the part should not exceed 1. |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(drapeForming_id3 is a DrapeForming)<br>and (drapeForming_id3 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of drapeForming_id3)<br>(maleMold_id8 is a MaleMold)<br>and (maleMold_id8 is hasTool of drapeForming_id3)<br>(part_id21 is a Part)<br>and (part_id21 is hasPart of design_id0)<br>(polymer_id1 is a Polymer)<br>and (part_id21 isMadeOf polymer_id1)<br><br>If<br>(hasDepth of part_id21 > 0)<br>(hasWidth of part_id21 > 0)<br>(hasDepth of part_id21/hasWidth of part_id21 > 1)<br><br>Then<br>feedback165 isNegative true<br>feedback165 has comment "For drape forming with a male mold, the depth-to-width ratio of the part should not exceed 1 .". |

Table F.1 (cont')

<table>
<tr><th>Id</th><th colspan="2">Results</th></tr>
<tr><td rowspan="3">105</td><td>Feedback</td><td>-Input sentence and the part needs revision:<br>Through-holes are preferable to blind holes because the pins forming them can be anchored at both ends and are therefore less susceptible to deflection during molding.<br><br>-Suggestion:<br>The clause possibly has unnecessary information. Consider enclosing the clause with parenthesis so that it is not processed.</td></tr>
<tr><td>Modified sentence</td><td>Through holes are preferable to blind holes (because the pins forming them can be anchored at both ends and are therefore less susceptible to deflection during molding).</td></tr>
<tr><td>Extracted rule</td><td>Given<br>(design_id0 is a MechnicalDesignService)<br>(blindHole_id8 is a BlindHole)<br>and (blindHole_id8 is hasFeature of hasPart of design_id0)<br>(rubber_id1 is a Rubber)<br>and (rubber_id1 is isMadeOf of hasPart of design_id0)<br><br>Then<br>feedback166 isNegative true<br>feedback166 has comment "Through holes are preferable to blind holes (because the pins forming them can be anchored at both ends and are therefore less susceptible to deflection during molding).".</td></tr>
</table>

Table F.1 (cont')

| Id | Results | |
|---|---|---|
| 107 | Feedback | -Input sentence and the part needs revision:<br>The minimum radius should exceed 1.5 times the external diameter .<br><br>-Suggestion:<br>The owner of the attribute <radius/external diameter> is not found. Consider revising the sentence or update domain ontology to specify the owner of the attribute. e.g. the attribute may be owned by other concepts such as... "<radius/external diameter> of the [part/feature]" |
| | Modified sentence | The minimum radius of a feature should exceed 1.5 times the outside diameter of the part. |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(part_id18 is a Part)<br>and (part_id18 is hasPart of design_id0)<br>(manufacturingFeature_id7 is a ManufacturingFeature)<br>and (manufacturingFeature_id7 is hasFeature of part_id18)<br>(rubber_id1 is a Rubber)<br>and (part_id18 isMadeOf rubber_id1)<br><br>If<br>not (hasRadius of manufacturingFeature_id7 > 1.5*hasOutsideDiameter of part_id18)<br><br>Then<br>feedback167 isNegative true<br>feedback167 has comment "The minimum radius of a feature should exceed 1.5 times the external diameter of the part .". |

Table F.1 (cont')

| Id | Results | |
|---|---|---|
| 109 | Feedback | -Input sentence and the part needs revision:<br>When parts are machined, outside radii should be greater than 1.5 .<br><br>-Suggestion:<br>The owner of the attribute <external radii> is not found. Consider revising the sentence or update domain ontology to specify the owner of the attribute. e.g. the attribute may be owned by other concepts such as... "<external radii> of the [parts]" |
| | Modified sentence | When a part is machined, its outside radii should be greater than 1.5. |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(mechanicalSubtraction_id6 is a MechanicalSubtraction)<br>and (mechanicalSubtraction_id6 is hasProcess of design_id0)<br>(part_id4 is a Part)<br>and (part_id4 is hasTargetPart of mechanicalSubtraction_id6)<br>and (part_id4 is hasPart of design_id0)<br>(ceramic_id1 is a Ceramic)<br>and (part_id4 isMadeOf ceramic_id1)<br><br>If<br>not (hasOutsideRadius of part_id4 > 1.5)<br><br>Then<br>feedback168 isNegative true<br>feedback168 has comment "When a part is machined, its outside radii should be greater than 1.5 .". |

Table F.1 (cont')

| Id | Results | |
|---|---|---|
| 110 | Feedback | -Input sentence and the part needs revision:<br>For dry-pressed parts, external edges should be beveled similar to that employed with powder-metal parts.<br><br>-Suggestion:<br>The phrase led by <similar/similarly> likely has unnecessary information. Consider enclosing the phrase with parenthesis so that it is not processed. |
| | Modified sentence | For dry-pressed parts, outside edges should be beveled (similar to that employed with powder-metal parts). |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(dry-pressing_id5 is a Dry-pressing)<br>and (dry-pressing_id5 is hasProcess of design_id0)<br>(part_id7 is a Part)<br>and (part_id7 is hasTargetPart of dry-pressing_id5)<br>and (part_id7 is hasPart of design_id0)<br>(outsideEdge_id9 is a OutsideEdge)<br>and (outsideEdge_id9 is hasFeature of part_id7)<br>(ceramic_id1 is a Ceramic)<br>and (part_id7 isMadeOf ceramic_id1)<br><br>If<br>not (outsideEdge_id9 isBeveled true)<br><br>Then<br>feedback169 isNegative true<br>feedback169 has comment "For dry-pressed parts, outside edges should be beveled (similar to that employed with powder-metal parts).". |

Table F.1 (cont')

<table>
<tr><th>Id</th><th colspan="2">Results</th></tr>
<tr><td rowspan="3">114</td><td>Feedback</td><td>-Input sentence and the part needs revision:<br>Diameter of machined holes should exceed 1.5, although smaller holes can be produced .<br><br>-Suggestion:<br>The clause possibly has unnecessary information. Consider enclosing the clause with parenthesis so that it is not processed.</td></tr>
<tr><td>Modified sentence</td><td>Diameter of machined holes should exceed 1.5, (although smaller holes can be produced).</td></tr>
<tr><td>Extracted rule</td><td>Given<br>(design_id0 is a MechnicalDesignService)<br>(mechanicalSubtraction_id4 is a MechanicalSubtraction)<br>and (mechanicalSubtraction_id4 is hasProcess of design_id0)<br>(hole_id6 is a Hole)<br>and (hole_id6 is hasTargetFeature of mechanicalSubtraction_id4)<br>and (hole_id6 is hasFeature of hasPart of design_id0)<br>(ceramic_id1 is a Ceramic)<br>and (ceramic_id1 is isMadeOf of hasPart of design_id0)<br><br>If<br>not (hasDiameter of hole_id6 > 1.5)<br><br>Then<br>feedback170 isNegative true<br>feedback170 has comment "Diameter of machined holes should exceed 1.5, (although smaller holes can be produced).".</td></tr>
</table>

Table F.1 (cont')

<table>
<tr><th>Id</th><th colspan="2">Results</th></tr>
<tr><td rowspan="3">121</td><td>Feedback</td><td>-Input sentence and the part needs revision:<br>For tubing and sections, the joint angle should not be smaller than 150.<br><br>-Suggestion:<br>The owner of the attribute <joint angle> is not found. Consider revising the sentence or update domain ontology to specify the owner of the attribute. e.g. the attribute may be owned by other concepts such as... "<joint angle> of the [part/sections/tubing]"</td></tr>
<tr><td>Modified sentence</td><td>For tubing, its joint angle should not be smaller than 150.</td></tr>
<tr><td>Extracted rule</td><td>Given<br>(design_id0 is a MechnicalDesignService)<br>(thermalWelding_id1 is a ThermalWelding)<br>and (thermalWelding_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of thermalWelding_id1)<br>(tube_id3 is a Tube)<br>and (tube_id3 is hasPart of design_id0)<br><br>If<br>(hasJointAngle of tube_id3 < 150)<br><br>Then<br>feedback172 isNegative true<br>feedback172 has comment "For tubing, its joint angle should not be smaller than 150.".</td></tr>
</table>

Table F.1 (cont')

<table>
<tr><th>Id</th><th colspan="2">Results</th></tr>
<tr><td rowspan="3">123</td><td>Feedback</td><td>-Input sentence and the part needs revision:<br>Avoid sharp internal corners, which concentrate heat-treating stresses.<br><br>-Suggestion:<br>The relative clause led by <, which> typically has unnecessary information. Consider enclosing the clause with parenthesis so that it is not processed.</td></tr>
<tr><td>Modified sentence</td><td>Avoid sharp internal corners(, which concentrate heat-treating stresses).</td></tr>
<tr><td>Extracted rule</td><td>Given<br>(design_id0 is a MechnicalDesignService)<br>(finishing_id1 is a Finishing)<br>and (finishing_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of finishing_id1)<br><br>If<br>(internalCorner_id4 isSharp true)<br>( (internalCorner_id4 is a InternalCorner)<br>and (internalCorner_id4 is hasFeature of hasPart of design_id0)<br>)<br><br>Then<br>feedback173 isNegative true<br>feedback173 has comment “Avoid sharp internal corners(, which concentrate heat-treating stresses).”.</td></tr>
</table>

Table F.1 (cont')

| Id | | Results |
|---|---|---|
| 125 | Feedback | -Input sentence and the part needs revision:<br>Avoid sharp edges and points, because paint tends to creep away from these areas and leave too thin film.<br><br>-Suggestion:<br>The clause possibly has unnecessary information. Consider enclosing the clause with parenthesis so that it is not processed. |
| | Modified sentence | Avoid sharp edges and points(, because paint tends to creep away from these areas and leave too thin a film). |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(finishing_id1 is a Finishing)<br>and (finishing_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of finishing_id1)<br><br>If<br>(edge_id4 isSharp true)<br>(point_id7 isSharp true)<br>(<br>( (edge_id4 is a Edge)<br>and (edge_id4 is hasFeature of hasPart of design_id0) ) or<br>( (point_id7 is a Point)<br>and (point_id7 is hasGeometry of hasFeature of hasPart of design_id0) )<br>)<br><br>Then<br>feedback174 isNegative true<br>feedback174 has comment "Avoid sharp edges and points(, because paint tends to creep away from these areas and leave too thin a film).". |

Table F.1 (cont')

| Id | | Results |
|---|---|---|
| 126 | Feedback | -Input sentence and the part needs revision:<br>Avoid blind areas, <u>i.e., surfaces that are behind corners or protrusions</u>.<br><br>-Suggestion:<br>The phrase led by <i.e.> typically has unnecessary information. Consider enclosing the phrase with parenthesis so that it is not processed. |
| | Modified sentence | Avoid blind areas<u>(, i.e., surfaces that are behind corners or protrusions)</u>. |
| | Extracted rule | Given<br>(design_id0 is a MechnicalDesignService)<br>(finishing_id1 is a Finishing)<br>and (finishing_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of finishing_id1)<br><br>If<br>( (blindArea_id3 is a BlindArea)<br>and (blindArea_id3 is hasFeature of hasPart of design_id0) )<br><br>Then<br>feedback175 isNegative true<br>feedback175 has comment "Avoid blind areas(, i.e., surfaces that are behind corners or protrusions).". |

Table F.1 (cont')

<table>
<tr><th>Id</th><th colspan="2">Results</th></tr>
<tr><td rowspan="3">132</td><td>Feedback</td><td>-Input sentence and the part needs revision:<br>The stroke should exceed 3 times the stock thickness.<br><br>-Comment<br>The owner of the attribute <stroke> is not found. Consider revising the sentence or update domain ontology to specify the owner of the attribute. e.g. the attribute may be owned by other concepts such as... "<stroke> of the [part]"</td></tr>
<tr><td>Modified sentence</td><td>The stroke of a letter should exceed 3 times the stock thickness.</td></tr>
<tr><td>Extracted rule</td><td>Given<br>(design_id0 is a MechnicalDesignService)<br>(marking_id1 is a Marking)<br>and (marking_id1 is hasProcess of design_id0)<br>and (hasPart of design_id0 is hasTargetPart of marking_id1)<br>(letter_id6 is a Letter)<br>and (letter_id6 is hasFeature of hasPart of design_id0)<br>(workpiece_id12 is a Workpiece)<br>and (workpiece_id12 is hasProcessInput of marking_id1)<br><br>If<br>not (hasStroke of letter_id6 > 3*hasThickness of workpiece_id12)<br><br>Then<br>feedback176 isNegative true<br>feedback176 has comment "The stroke of a letter should exceed 3 times the stock thickness .".</td></tr>
</table>