Development of a Knowledge Base using Human Experience Semantic Network for Instructive Texts

by

Sk Sami Al Jabar

A thesis submitted to the
School of Graduate and Postdoctoral Studies in partial
fulfillment of the requirements for the degree of

Master of Applied Science in Electrical and Computer Engineering

Faculty of Engineering and Applied Science
University of Ontario Institute of Technology (Ontario Tech University)
Oshawa, Ontario, Canada
December 2021

THESIS EXAMINATION INFORMATION

Submitted by: **Sk Sami Al Jabar**

Master of Applied Science in **Electrical and Computer Engineering**

Thesis title: Development of a Knowledge Base using Human Experience Semantic Network for Instructive Texts

An oral defense of this thesis took place on December 17, 2021 in front of the following examining committee:

Examining Committee:

Chair of Examining Committee:	Dr. Khalid Elgazzar
Research Supervisor:	Dr. Hossam Gaber
Co-Research Supervisor:	Dr. Jing Ren
Examining Committee Member:	Dr. Sanaa Alwidian
Thesis Examiner:	Dr. Ramiro Liscano, Ontario Tech University

The above committee determined that the thesis is acceptable in form and content and that a satisfactory knowledge of the field covered by the thesis was demonstrated by the candidate during an oral examination. A signed copy of the Certificate of Approval is available from the School of Graduate and Postdoctoral Studies.

Abstract

An organized knowledge base plays a vital role in retaining knowledge. Instructive text (iText) consists of a set of instructions to accomplish a task or operation. In the case of iText, storing only entities and their relationships is not enough for capturing knowledge from iTexts. iTexts consists of parameters and attributes of different entities and their actions based on different operations. The values differ for every operation or procedure for the same entity. As a result, existing approaches created limitations in capturing knowledge from iTexts. This research presents a knowledge base for capturing and retaining knowledge from iTexts existing in operational documents. From each iTexts, small pieces of knowledge are extracted and represented as nodes and edges in the form of a knowledge network called the human experience semantic network (HESN). The knowledge base also consists of domain knowledge having different classified terms and key phrases of the specific domain.

Keywords: knowledge-base; natural language processing; human experience semantic network; entity relationship extraction; knowledge representation

Author's Declaration

I hereby declare that this thesis consists of original work of which I have authored. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I authorize the University of Ontario Institute of Technology (Ontario Tech University) to lend this thesis to other institutions or individuals for the purpose of scholarly research. I further authorize University of Ontario Institute of Technology (Ontario Tech University) to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research. I understand that my thesis will be made electronically available to the public

_____ Sk Sami Al Jabar

Statement of Contribution

This research has led to the following publications, directly extracted from this thesis:

1. Hossam A. Gabbar, Sk Sami Al Jabar, Hassan A. Hassan, and Jing Ren. Development of knowledge base using human experience semantic network for instructive texts. Applied Sciences, 11(17):8072, August 2021. [20]

2. Hossam A. Gabbar, Sk Sami Al Jabar, Hassan A. Hassan, and Jing Ren. An intelligent experience retention system: Challenges and limitations for operation and maintenance in nuclear power plants. IEEE Syst., Man, Cybern. Mag., 7(4):3134, October 2021. [21]

Acknowledgements

This work would not have been possible without the years of guidance and encouragement of my research supervisor Dr. Hossam Gaber and co-supervisor, Dr. Jing Ren, both from the Ontario Tech University (University of Ontario Institute of Technology - UOIT). They have been an inspirational mentor, and I am very fortunate to have had the pleasure of working with them. Furthermore, I am thankful to the thesis examiner and the thesis committee for helping me to improve and elaborate the explanation of the thesis work.

I am also thankful to Natural Sciences and Engineering Research Council of Canada (NSERC) and Ontario Power Generation (OPG) for funding this research.

Special appreciation to Dr. Hassan A. Hassan from OPG for contributing his knowledge, expertise and always being there alongside our research work.

Dedication

This is dedicated to my parents namely Sk Alauddin and Papia Begam, my elder brother Sk Tajbir Boney and to my dear wife Afia Noshin.

Table of Contents

List of Figures

Chapter 1

Introduction

In this chapter, the research motivation, industrial challenges and the problem definition have been discussed. The objectives of the thesis have also been mentioned in this chapter. This will lead to the end goal of solving the problem and filling up the gap. Moreover, this chapter also includes a summary of the entire methodology. Finally, the outline of the whole thesis has also been added for easy understanding of the reader.

1.1 Research Motivation

The research motivation behind this thesis is based on a real-world problem. The proposed technique deals with instructive text (iText), and captures knowledge from

iTexts. iTexts are the set of texts that consist of a title or name of any operation and few instructions or procedures on how to successfully accomplish that operation. iTexts, found in industrial and operational documents, are followed to accomplish different operations and tasks. These documents are large in quantity and consist of 50 to 100 pages in each document. As a result, it is time-consuming and risky to go over these documents, retrieve desired instructions and perform accordingly during a critical operation or task. If all these operational instructions could be extracted as knowledge and structured in a knowledge base, then the knowledge base would guide the workforce. It would help accomplish complex tasks by providing desired information based on different operations and instructions when asked. Hence, our proposed knowledge base would retain all the necessary information and instructions from these documents. This knowledge base can be used to retrieve desired queries later on. It will help save time, reduce the training period of new employees, and less experienced workforce will also be able to perform comparatively more critical tasks than before. The knowledge base will also help reduce the industry's risk by providing appropriate information when desired based on different parameters. Whereas humans may make mistakes by retrieving wrong information from the document, especially the less experienced employees, and this may lead to operation failure. Not only this but iTexts are being used in many other domains as well. Existing knowl-

edge bases or knowledge representation techniques created limitations in capturing and retaining knowledge from iTexts. Establishing relationships among different entities or terms is not enough. Each of these relationships must be structured in a way so that it is clear from which operation or instruction the relation between the terms got established. The relationships between two terms in the case of iTexts are based on different operations and instructions. As a result, it is crucial to retain this information. A structured knowledge representation would help to do complex reasoning in future. And this gave me the motivation to work on this research. However, this thesis demonstrates only the technique of structuring the knowledge from iText.

1.2 Background

Instructive texts (iTexts) are different, in terms of structure and textual pattern. iTexts usually instruct or describe how to do something in a step-by-step process [25]. For example, how does one fix a turbine? The answer to this question has a few procedures to follow, which will help accomplish the main goal or operation. iTexts usually consist of a title, which could be the name of the process or operation, and a set of instructions or procedures that help to accomplish the operation in a step-by-step process. Figure 1.1 shows the differences between regular or standard text and iText in terms of structure and textual pattern.

Figure 1.1: Difference between regular text and instructive text (iText)

Employees in large industries, such as the nuclear industry, store information like procedures, precautions, experiments, risk factors, etc., in handwritten or pdf documents, which are prominent in quantity. These are called operational documents. They follow these documents during operation in order to accomplish each task efficiently. There is a continuous movement of experienced personnel to different departments, or they go for retirements and hence a tremendous amount of expertise is lost. The loss of expertise costs the industry a huge amount of money as they have to invest in training less experienced personnel, leading to indirect losses in delayed or wrong activities. A less experienced employee cannot operate complex tasks due to having less knowledge and training about the documents and their operation. The training period could take months to cover information about the different operations. The more extended the training period, the more expensive it is for the industry. At many

times, it is troublesome to retrieve any specific information during operation or other practices. It is helpful if the desired information is quickly retrieved when employees are in the middle of an industrial activity or in a lab, making them work faster. Moreover, much time is wasted while searching for specific information from one out of innumerable documents during a complex operation to accomplish its objective. In case of any inaccurate information retrieval, there is a high risk of operational failure, which is again costly to recover for the industry. If information and human experience from these large number of documents could be extracted, structured and retained in a knowledge base from where desired information could be easily retrieved at any time, then the operational time could be saved and utilized in a much better way. Furthermore, this could also reduce the expenses for the training and learning purposes. The learning process could also be faster. The less experienced employee will also be able to perform the complex operation with the help of the knowledge base, which was impossible for them previously. However, the management of this knowledge base could be critical with the increase in information. Without proper structuring of knowledge, information retrieval will be an expensive approach.

Hence, this knowledge can be structured in the form of a network, being able to retain the human expertise from these documents in an organized way by developing relationship among the entities, their actions, attributes and different values and

parameters from each of the iTexts and procedures, which could have information about human role, tool, equipment, location, document, operation, procedure, etc., associated with that particular operation. Current research approaches in developing a knowledge base and retaining the relationship among entities and structuring knowledge from standard texts does not fully apply for iTexts as relationships, attributes, and properties of entities in iTexts differ in case of different operation. This thesis presents a knowledge base consisting of HESN and domain knowledge. HESN structures knowledge by capturing the human experience from iTexts which includes relationships among different terms, information about the operations and instructions. The domain knowledge consists of classes and properties of information related to any specific domain. The knowledge base retains the properties, relationships and values of different entities, action terms or verbs, attributes, and attribute values found in an iText for different operations. It extracts the real expertise from iTexts and dynamically updates the HESN existing in the knowledge base. The contribution of this work can be summarized as follows:

1. The development of an adaptive, dynamic and deterministic knowledge structure with qualitative and quantitative attributes, called the human experience semantic network (HESN), is used to capture and structure knowledge from iTexts in the form of nodes and edges;

2. The development of a knowledge base, consisting of HESN and domain knowledge, for retaining properties, values, and relationships of different terms or key phrases, found in iTexts. These terms or key phrases could be an entity, action term or verb, attribute, or attribute value. The knowledge is structured for different entities, action terms, attribute, or attribute values based on different operation.

3. The development of a learning algorithm which helps to establish relationship among different terms or key phrases based on different operations and instructions, found in the iText, with the help of the domain knowledge and form HESN;

1.3 Problem Definition

Different approaches for Knowledge extraction and knowledge base development for regular texts has created limitation in capturing and retaining the human experience from instructive texts (iTexts), consisting of operational procedures. Entity Relationship extraction [38], ontology based knowledge extraction [34], knowledge graph [72] or knowledge base development [35, 73] and similar previous approaches lack in establishing relationship among different entities, action and parameters found in iTexts

based on different operation. The relationship among different terms varies based on different operations and instructions. This needs to be dynamically structured, without which it is hardly possible to learn and extract knowledge from iTexts. This leads to failure in dynamically providing answers to questions related to any operational procedure when asked.

1.4 Research Objective

There are two major objectives in order to reach to the final goal of successful development of knowledge base consisting of HESN and domain knowledge. Under each of these objectives there are tasks that are completed to satisfy the end result of the objective. Apart from these, a prototype is established in order to test the tasks and run the algorithms accordingly.

1.4.1 Objective 1: Development of the Knowledge Base using HESN and Knowledge Domain

The first research objective includes the development of the knowledge base consisting of Human Experience Semantic Network (HESN) and domain knowledge. This leads to two main sub tasks which are as follows -

1. Development of HESN which is an adaptive, dynamic and deterministic knowledge structure having qualitative and quantitative attributes. It is used to capture and structure knowledge from iTexts in the form of nodes and edges.

2. Development of the Domain Knowledge that consists of domain specific words or phrases which could be any one of the following types - entity, action, attribute or attribute value. Each of these words or phrases belong to a class and have properties based on the category of class.

3. Validation of the knowledge base which includes HESN and Domain Knowledge.

1.4.2 Objective 2: Development of learning algorithm based on HESN

The second research objective includes the development of the iText learning methodology from iTexts and dynamically updating the HESN. The following sub tasks help to achieve the goal –

1. Development of an algorithm which helps to detect and establish relationship among any entity, action, attribute or attribute value, found in each iText, based on the domain knowledge, in the form of duplet. These duplets together

form a small network for each iTexts. This small network is the building block of HESN.

2. Development of a methodology based on a popular Natural Language Processing technique called Parts-of-Speech (POS) tagging. This methodology helps to generate tags against each relationship among different terms or phrases, to structure knowledge about different entities, action, attribute or attribute values based on different operations.

1.5 Proposed Methodology

The knowledge base proposed in this research captures knowledge from iText. iText consists of an operation title and a set of instructions or procedures that talk about how that particular operation may be accomplished successfully. The knowledge base is composed of two components.

The first component is domain knowledge. It consists of all the domain-specific terms and phrases defined by a domain expert manually. These terms and phrases are classified to know which one is a tool, equipment, human role, location, etc. All these classes are divided into four major categories — (1) Entity or Name Phrase, (2) Action, (3) Attribute and (4) Attribute Value or Value. Each class belongs to a

category. Under each class, there are many terms defined according to the plan of the domain expert. The properties of the classes are the same based on each category. For example, all classes under the category of Entity will have the same properties and sub-properties. The domain knowledge is used to identify different terms and phrases found in the iText, and the HESN is generated later on.

The second component is a knowledge network called Human Experience Semantic Network (HESN), which represents the relationships of different entities, action terms, attribute terms, and values based on different operations. This is the main component of the knowledge base. The representation of relationships is done in the form of nodes and edges. These relationships are established and captured from iText based on different operation titles, with the help of our approach. For e.g. a relation may be found among 'reactor lab,' 'wear,' and 'helmet' in an iText mentioned in 'Operation A.' For the same entity 'helmet,' there could be another relation among 'helmet,' 'wear,' and 'construction zone' captured from another iText mentioned in 'Operation B.' This means for the same entity 'helmet,' its relations vary based on two different operations. The same goes for the action term 'wear' as well. This operation-based relationships establishment process is performed with the help of tags extracted as nouns and verbs from each iText with the help of the Parts-Of-Speech tagging process in Natural Language Processing. Against each relation, these

tags are used, which separates the relations of a word or phrase made with others based on different operations.

HESN consists of relationships among different terms. A term could be belonging to a class under the category of either Entity, Action, Attribute or Value. Hence, there could be a relation between an Entity and an Action, or Action and Attribute and so on. Every term is represented as a node and is related to another term with the help of an edge. This edge represents anyone out of six types of relationships. These are— i) entity-action (E-Ac), (ii) entity-entity (E-E), (iii) entity-attribute (E-Att), (iv) entity-value (E-V), (v) action-attribute (Ac-Att), and (vi) attribute-value (Att-V). Suppose a relationship between a term that falls under a class categorized as action in the domain knowledge and another term similarly categorized is found. That is, an Action-Action (Ac-Ac) relationship is observed. In that case, it is considered invalid, and that relation is not captured by HESN. The goal is to keep the relationships among the terms meaningful. Establishing relationship between two action terms does not make much sense. For example, if 'sign' and 'move' are related where both are action terms, it does not make much sense. Same goes for 'run' and 'high' where the term 'run' is an action(Ac) and the term 'high' is a value (V).

A proposed algorithm that detects terms or phrases based on domain knowledge has been proposed. It then creates relationships in the form of duplets among different

terms, phrases, or even numbers, generates tags against each relation, and updates the knowledge base by adding new relations into the HESN. The algorithm helps to read and learn from iTexts about what type of terms are related to what other terms based on different operations and instruction. The iText is learned and retained as knowledge in HESN. This is a learning process since, firstly, the HESN retains information about the type of relationships found between every two terms in any particular instruction of an operation. Secondly, the semantics of the terms are also retained in HESN since each of the terms is classified, and these classes provide meaning to each term. Moreover, the more iTexts are read, the more new relationships are formed. The result of all these new relationships are the outcome of learning with the help of the proposed algorithm.

1.6 Thesis Outline

Chapter 1 consists of research motivation, background, problem definition, research objectives, and the summary of our proposed methodology. Chapter 2 discusses research works related to knowledge-based and ontology-based approaches. It also consists of research work related to entity-relation extraction approaches from texts. Chapter 3 explains our proposed methodology in detail. A case study has been demonstrated in chapter 4. This case study helps to understand the methodology.

In chapter 5, the advantage of our proposed approach is explained. In chapter 6, an industrial implementation is demonstrated where a software system is used, which uses a part of our research concept. Finally, in chapter 7, research contribution, limitations, future work, and publications have been discussed.

Chapter 2

Literature Review

There are many ways to capture knowledge from text and develop a knowledge base, knowledge network, or knowledge graph to represent the acquired knowledge. This section discusses research work related to the knowledge base, information extraction approaches, and entity-relationship establishment approaches to provide information about recent work on how knowledge is acquired and represented from different kinds of texts based on different domains. One major part of our proposed knowledge base consists of information about the relationships of different terms and key phrases in iText based on different operations. All these relations are represented in the knowledge network called HESN. Hence, the literature review includes research works on both knowledge base techniques and entity-relationship extraction techniques to

represent knowledge.

2.1 Knowledge-Based Approach

The system, which is based on a knowledge base, consists of information and data structured in an organized way. Question answering over a developed knowledge base based on the domain knowledge helps retrieve the information as demanded through a query. It has been mentioned in [4] - "In the paradigm of Knowledge-Based Systems (KBS), the design of methods to simplify the reasoning leads to more efficient processes." Various knowledge-based approaches, such as semantic networks [49], Bayesian networks [69], fuzzy rules [70], fuzzy cognitive maps (FCMs) [45], case-based reasoning (CBR) [9], and association rule mining (ARM) [15, 28] have been proposed in order to establish intelligent knowledge-based systems. It is necessary to develop a dynamic knowledge base that can capture knowledge and dynamically update the knowledge base in such a case. Knowledge-based approaches are also used for learning and extracting knowledge from texts based on a specific domain.

In [72], a method has been proposed for the development of a knowledge base based on the knowledge and behavior of operators in terms of a severe accident in nuclear power plant. The knowledge base was developed using 281 scientific publi-

cations, which were summarized and then the knowledge was extracted from them. The publications were related to the terms "severe accident" and "nuclear". The knowledge base that constitutes the knowledge graph consists of nodes and edges representing the causal relationships, entities, states, and affiliations. A knowledge graph was generated from each publication summary, and all these knowledge graphs were merged to constitute one main knowledge graph. From this methodology, it is observed that texts are skipped because it is considered repetitive or irrelevant to the topic of interest. This skipping of text is unacceptable in dealing with iTexts as each and every information in each of the instructions is required to be extracted and structured correctly in the knowledge base. Moreover, procedural knowledge structuring, information tracking, and sequencing are a significant part of structuring knowledge from iTexts which cannot be solved only with entity relationships. In [53], an agricultural knowledge base or framework has been proposed which helps to identify pests and diseases that affect a crop. An automatic ontology population tool has been developed. It helps extract relevant data from unstructured documents with the help of natural language processing techniques and update ontology. Their approach included representing information related to symptoms of plant diseases based on plant parts and damages. The proposed methodology was built into a system that could recognize crop pests with the help of the knowledge base. A sys-

tem called "Smart Farming" was proposed in [55] for precision farming management, where a knowledge base was developed. Information was organized in the knowledge base in the form of a semantic network consisting of concepts and relations. The knowledge represented in the knowledge base was about crop production, production resources, agricultural machinery, equipment, and other resources. Ontological principles were adopted to design the domain model based on concepts, attributes, and interrelations. In [52], a knowledge-based strategy has been proposed for data management and mining in machining and to support decision making. The knowledge base consisted of manufacturing knowledge and a multi-level model that helped acquire knowledge and decision-making from the information stored in the knowledge base. Operation optimization knowledge base system (OOKBS) was designed in [73] for the operation optimization of a polyethylene process. Knowledge was represented using an ontology. Knowledge of polyethylene process, equipment operation, and operation optimization were integrated into the knowledge base. A neural network model was developed to identify the relationship between operating conditions and molecular weight distribution (MWD) parameters.

Knowledge base is not a new concept. Previously, thesis work has been done on representation of knowledge [35, 47, 11, 66, 42, 3] which helped in capturing knowledge based on particular domains. Moreover, there are innumerable research papers where

knowledge bases were implemented for different domain [2, 12, 19, 10, 46, 17, 29, 41] in order to solve different problems and expound unclear challenges. In manufacturing and production, the decision-making processes are performed by humans and their knowledge-based processes [48]. An established Knowledge base helps to retain knowledge and do reasoning in order to reach to a conclusion. As a result, its formation and arrangement needs to be well structured as well as expandable or agile.

2.2 Ontology-Based Approaches

Ontology-based approaches are very popular and essential approach, which help in the representation of acquired knowledge. They are used for data and knowledge integration. Ontology-based approach facilitate question answering and reasoning over data. Formal ontology engineering is considered as a task which is difficult to perform, very much time consuming and high costs [5]. Ontology learning helps to solve this problem, where ontology is automatically generated. That means, in ontology learning, conceptual knowledge is extracted from input and ontology is built from them [31]. Some techniques required building ontology from scratch, while others use existing ontologies [64].

In [34], an ontology-based approach was proposed, which classifies security re-

quirements automatically. The security requirements were described with the help of 35 defined linguistic rules and 140 defined security keywords. The security requirements ontology was defined using description logic (DL). All these are used to train classifiers of security requirements using machine learning algorithms like naïve Bayes (NB), decision tree (DT), and logistic regression (LR). In [65], an ontology, named concrete bridge rehabilitation project management ontology (CBRPMO), was presented, which was developed using domain knowledge of bridge rehabilitation and following standard procedures. Semantic reasoning rules were constructed to support dynamic information integration and management functions. The developed ontology aims to investigate the information in bridge rehabilitation projects and efficiently support constraint management. The ontology was developed using web ontology language (OWL).A knowledge-based model for additive manufacturing (AM) has been proposed in [54] using ontology, where data are organized with the help of the ontology structure. A form is filled up with data and based on that, data validation and reasoning are done with the help of associated rules that determine the appropriate machine name or model that can do the manufacturing. Moreover, the paper [63] proposes a knowledge-based approach that covers different ontology learning methods from the text.

2.3 Entity-Relation Extraction

In natural language processing, entity and their relationship extraction is considered as a basic task of information extraction [32, 59, 61], and it can help to accomplish a range of tasks, where knowledge base construction [24, 37] is one of them. The main objective of the entity recognition and relation extraction task is to determine the relational structure of the mentioned entities from unstructured texts. The task has two subtasks — (i) named entity recognition (NER) [43] and (ii) relation extraction (RE) [6]. These tasks help to connect each entity with other ones and are very useful for developing a semantic network with nodes and edges. Several semantic relation extraction method has been proposed, which can be divided into four categories — (i) supervised [71], (ii) unsupervised [36], (iii) distant supervision [40], and (iv) semi-supervised [14]. In this research work, our knowledge base help to identify the entity and key phrases from iTexts and represent the relationship among them based on different operation in the form of HESN.

In [38], an approach has been proposed for constructing knowledge graphs with the help of a task named relation extraction and linking. Their approach is dependent on information extraction (IE) tasks for obtaining the named entities and relations. Finally, these are linked using data and standards of the semantic web. Initially, the

input text is transformed into resource description framework (RDF) triples using the combination of natural language processing and information extraction operation. The information extraction operation includes tasks like document acquisition and preprocessing the input text, extracting named entities and their association with the grammatical unit, semantic relation extraction using the OpenIE approach and associating it with semantic information provided by an approach termed as semantic role labeling (SRL) that helps to identify the order and selection of elements which are to be finally represented through RDF triples. In total, 605 IT news webpages were downloaded and used for the evaluation of their research methodology. It had about 12,015 sentences which were processed to construct the RDF statements. RDF triples or statements are dependent upon the subject, object, and predicate of a sentence. Many sentences were ignored that had relations containing no named entities in subject and object. As a result, RDF statements for such sentences were not created and, thus, ignored. In [50], relation extraction was done using entity indicators. These entity indicators were inserted into each relation sentence and this helps the neural network know about the position, syntactic and semantic information of the named entities in the relation sentence. This approach help to solve the problem related to having several named entities in a sentence where relation extraction is complex. Task-related entity indicators were designed in their research for the neural

network to know the position information of named entities. In [30], a relation extraction method for construction of COVID-19 information knowledge graph based on deep learning was proposed. It is another open information extraction (OpenIE) system based on unsupervised learning without any pre-defined dataset, although a COVID-19 entity dictionary was created and used for scraping related information. The proposed method extracts knowledge from documents consisting of information related to COVID-19 and constructed a knowledge base that consisted of connecting words between COVID-19 entities, which was captured from COVID-19 sentences. The proposed model could identify a relation between COVID-19-related entities using (BERT), and it does not need any pre-built training dataset. In [7], researchers presented a neural model to extract entities and their relation from texts. The basic layers of their proposed model consisted of embedding layer, bidirectional sequential long short term memory (BiLSTM) layer, conditional random fields (CRF) layer, and sigmoid layer. The conditional random fields (CRF) layer was used to recognize entities and the sigmoid layer for entity relation extraction. The task was modeled as a multi-head selection problem where an entity may have multiple relations in a text. The model does not rely on hand-crafted or external natural language processing tools, such as parts-of-speech (POS) tagger, dependency parsers, etc. Extracting semantic relation from text has been performed by a group of researchers in [62]. Two models

named Rel-TNG and Type-TNG were proposed that used topic n-Grams (TNG). These two models were able to show similar performance measure for Rel-LDA and Type-LDA, but the models outperformed Rel-LDA and Type-LDA when there was prior knowledge available. GENIA and EPI datasets were used for this experiment which are biomedical texts. Two types of relationships were annotated: PROTEIN-COMPONENT and SUBUNIT-COMPLEX. One of their advantages was that these annotations were already done and were provided with the dataset. Ref. [44] shows another knowledge graph construction mechanism that involves entity-relationship establishment from triples consisting of entities and their relationships and also fulfilling relationship gaps between entities from texts containing those relationships. The approach uses texts to fulfill relationship gaps found between entities. The knowledge graph does not capture the relationship between texts as their approach is not aimed to capture that. They aimed to find and identify triples (h, r, t), where h and t are entities, and r is the relationship, and extract relationship from there and use texts to find if it is missing between any h and t. An investigation is done in [67] which narrated the influence of semantic link networks on the performance of the question answering system. It is accomplished by enhancing the ability of the system in answering different types of questions and supporting different patterns of answering questions with the help of the semantic link network. The accuracy of an answer

against a question depends widely on the answer range and the number of semantic links on the answer range. By answer range, it has been meant to have more texts having potential answers. The research work clarifies that the greater the number of semantic links there is, the accuracy and formation of the answer will be better against a question. The semantic link network is formed from semantic objects and their semantic links that connect two semantic objects. These objects consist of a form of a string with their synonyms. Semantic link network, produced by the researchers, connects different terms from a range of text and establishes relationships, which is almost similar to entity relationship establishment.

2.4 Limitations in Case of iText

Different methods and approaches of information extraction and knowledge structuring discussed are suitable for learning from regular or standard texts or paragraphs consisting of information about different entities and their relationships. Relationships among entities were established in the form of RDF triples, semantic networks or knowledge graph. Knowledge extraction is performed from texts and developed knowledge-base. Ontology is developed for better success of the knowledge extraction process. Semantic meaning is extracted with the help of ontology. Identification of different problems is performed based on established knowledge-base or ontology.

Such approaches have limitations in capturing knowledge from iTexts that consists of a set of instructions related to how to conduct an operation or activity. The structure of the sentence in iText is a bit different from regular paragraphs. Firstly, in iTexts, there could be an entity having different values or relationships based on different operations. Hence, it is essential to keep track and structure knowledge of the values and relationships of an entity based on different operations. Secondly, there could be relationships between two entities and between an entity and other terms like "move", "shift", "high", "low", or any number. Traditional triplets extraction or RDF triples are extracted from sentence structure consisting of subject, object, and predicate. This method is not perfect as it sometimes consists of error or contradictory information [23]. Moreover, having more than 1 triple in a sentence is again required to be handled which is also an expensive process [16]. Predicting triple in a sentence is also another task that needs effort. It is also not always possible to get information in the form of triplets. In Figure 1.1, the iText "Wear gloves" consists of two words only. Here, a particular user could be considered as subject. However, in case of iText, this is not always applicable. Considering "user" as subject and relating this entity with other two terms will create confusion in the case where this "user" is already defined as any particular human role in the operation title. Therefore, if the operation title is "Must perform tasks for Lab Operators", then wearing gloves is in-

structed for lab operators instead of "user". Duplet based relation extraction helps to create relation between the action "wear" and entity "gloves". This piece of information also consists of tags from operation title that helps to know that wearing gloves is applicable for lab operators. This tracking of information is explained in more detail in methodology part of this thesis. Moreover, ontology or domain knowledge in case of RDF needs to be enriched. However, in our case, the domain knowledge is developed in a simple way. It consists of different class names and words or phrases defined under each class name, which is good enough for generating duplet relations. Further explanation about domain knowledge is done in the methodology part of this thesis. Another example can be drawn when numbers are considered. In case of duplets, our approach identifies number and can make relation with another term or phrase directly, whereas, it is a complicated task when triplets are considered. For example, "Pump must have pressure 4 Pa". From here, we get the duplets (pump, pressure), (4, pressure). Here, the value 4 is directly assigned with pressure which will be a helpful information for complex reasoning when this operational instruction is considered. This is not possible with triplets directly. For this reason, our research deals with duplets. The domain knowledge also needs to be well defined. This will help to identify different terms found in iTexts and structure knowledge accordingly. This thesis proposes a knowledge base that captures knowledge from iTexts, repre-

Existing Approaches	Limitations
Knowledge Based Approach	• Do not retain information about from which text the relationship got established. • Text skipped if relationship is not found or repeated.
Ontology Based Approach	• Complex domain knowledge definition or ontology structure.
Entity Relationship Extraction	• Existing approaches lack in establishing relation between an entity and number.
RDF Triples	• RDF triples are not formed if text does not have subject, object and predicate. • Establishing relationship among entities where more than 1 triples are present is an expensive process. • RDF triples sometimes consists of error or contradictory information.

Figure 2.1: Summary of limitations of popular existing approaches in the case of iText

sents the knowledge using HESN as part of the knowledge base, and dynamically updates the knowledge base. The limitations of the popular existing approaches has been mentioned in figure 2.1

Chapter 3

Research Methodology

The development of the knowledge base for iTexts is a step-by-step process. The two major parts of the knowledge base are domain knowledge and HESN itself. They help accomplish tasks, such as identifying different terms and key phrases, establishing relationships among them, structuring knowledge of different terms and key phrases found in iTexts based on different operations under which each of the instructions is provided and finally update HESN and the knowledge base. For simplicity, all entities and named entities are termed entities in this thesis. In this research, a relationship is established among four types of terms or key phrases—entities, action terms or verb terms, attribute terms, and attribute values. Domain knowledge consists of information related to these terms or key phrases. They are represented using class

and property and help to detect and identify terms and phrases in iTexts. Each time new instructions are learned, and the HESN is updated. Updating HESN or domain knowledge also means updating the knowledge base since HESN and domain knowledge constitute the knowledge base.

3.1 iTexts Extraction and Preprocessing

This research is done based on the test documents communicated with the maintenance section within Ontario power generation (OPG), responsible for approximately half of the electricity generation in the Province of Ontario, Canada. The test documents had different contents related to purpose, pre-requisites, instructions, post-requisite, definitions, summary of changes, validation, and verification, and similar information about different processes, operations, inspections, equipment, etc. The sections in the document that consisted of operational procedures and instructions, only those sections were extracted. This is done with the help of an algorithm or parser which follows the structure and pattern of the documents and finally groups the iTexts in the appropriate section of the document. In this way, the entire document is divided into small chunks where each chunk consists of the title of the operation or procedure (Parent-iText or PT) and a set of instructions underneath (Child-iText or CT). These groups or chunks of texts were further processed to capture knowledge

and retain it in the knowledge base with the help of HESN and domain knowledge. However, this thesis is focused mainly on extracting knowledge from iTexts and developing the knowledge base and HESN rather than text extraction from documents.

3.2 Domain Knowledge Development

The domain knowledge is an essential part of the knowledge base. The domain knowledge consists of different terms and phrases which are classified as 'resource', 'equipment', 'humanRole', 'document', 'action', 'attribute', 'attributeValue' etc. Different terms and phrases are classified under different classes, and the work is done by a domain expert manually before any iText is even read. For e.g. there could be a term 'manager' classified as 'humanRole', 'move' classified as 'action', 'pressure' classified as 'attribute', 'compactor' classified as 'equipment', 'high' classified as 'attributeValue' and so on. All these classes are divided into four major categories — (1) Entity or Name Phrase, (2) Action, (3) Attribute and (4) Attribute Value or Value.

1. Entity or Name Phrase: Under this category, there could be different types of classes, namely 'resource', 'equipment', 'location', 'tool', 'humanRole' etc. The properties of all these classes are the same. The terms found under this class are mostly nouns or names of different elements, person, location etc.

Class Category	Example Class Names	Example Terms
Entity	Resource	Pump, Fuel, Coolant etc.
	HumanRole	Manager, Personnel etc.
	Document	Record Sheet, Appendix A etc.
Action	Action	Move, Configure, Repair etc.
Attribute	Attribute	Pressure, Height, Weight etc.
Value	AttributeValue	High, Low, Warm, Dry etc.

Figure 3.1: Example of different terms are classified and categorized in the domain knowledge.

E.g. suit, appendix B, personnel, FLM (First Line Manager), signature record sheet etc. The properties belonging to this class are 'Name', 'AssociatedAction', 'AssociatedAttribute', 'AssociatedEntity' and 'AssociatedValue'. Each of these properties again has four sub-properties. These are 'RelatedTo' (it consists of the name of the other term with which the current term is associated), 'PT' (consists of the parent text, which is the title of operation), 'CT' (consists of the child text, which is a particular instruction under an operation, where the relationship with a particular term is established) and 'Tags' (which are the combination of nouns and verbs extracted from PT and that particular CT). The values of all these properties, except for the property 'Name', are not

assigned initially. That means, when a domain expert assign different terms to different classes, the terms get all these properties and sub-properties based on the category of class. But the value of the properties is empty, except for the property 'Name', which consists of the name of the term or phrases. The values of the properties and sub-properties are dynamically updated when an iText is learned. These values help to know how terms are related to each other, from which iText (CT) the relationship was captured, the operation title (PT) under which the iText (CT) was found and finally, the tags;

2. Action: Under this category, there is only one class which is also called 'action'. It again has a set of properties. The terms found under this class are mostly verbs. E.g. wear, sign, verify etc. The properties belonging to this class are 'Name', 'AssociatedAttribute' and 'AssociatedEntity' and similar sub-properties for each, as observed for classes under the Entity or Name Phrase category;

3. Attribute: Under this category, there is only one class which is also called 'attribute'. The terms found under this class could be channel number, start time, date, height, body condition etc. The properties belonging to this class are 'Name', 'AssociatedAction', 'AssociatedEntity' and 'AssociatedValue' and

similar sub-properties for each, as observed for classes under the Entity or Name Phrase category;

4. Attribute Value or Value: Under this category, there is only one class which is also called 'attributeValue'. It could have terms like high, low, in progress etc. Moreover, any number also belongs to this category, but these numbers are not pre-defined. Whenever a number is detected in an iText, it is considered as an attribute value. The properties belonging to this class are 'Name', 'AssociatedAttribute' and 'AssociatedEntity' and similar sub-properties for each, as observed for classes under the Entity or Name Phrase category;

These are the four major categories, and the floor is open for the domain expert to design and categorize classes in whichever way they want and assign different terms under those classes based on the four different categories. An example of a few classes under different categories and a few terms under different classes has been shown in figure 3.1. All these terms help to detect different terms from iText, and that also gives an idea about what category of terms are present in that particular iText. Figure 3.2 represent three out of many terms from the domain knowledge. Again, each of these terms could be considered as a disconnected node. Before reading any iText, when domain experts assign different terms and phrases with different classes, each

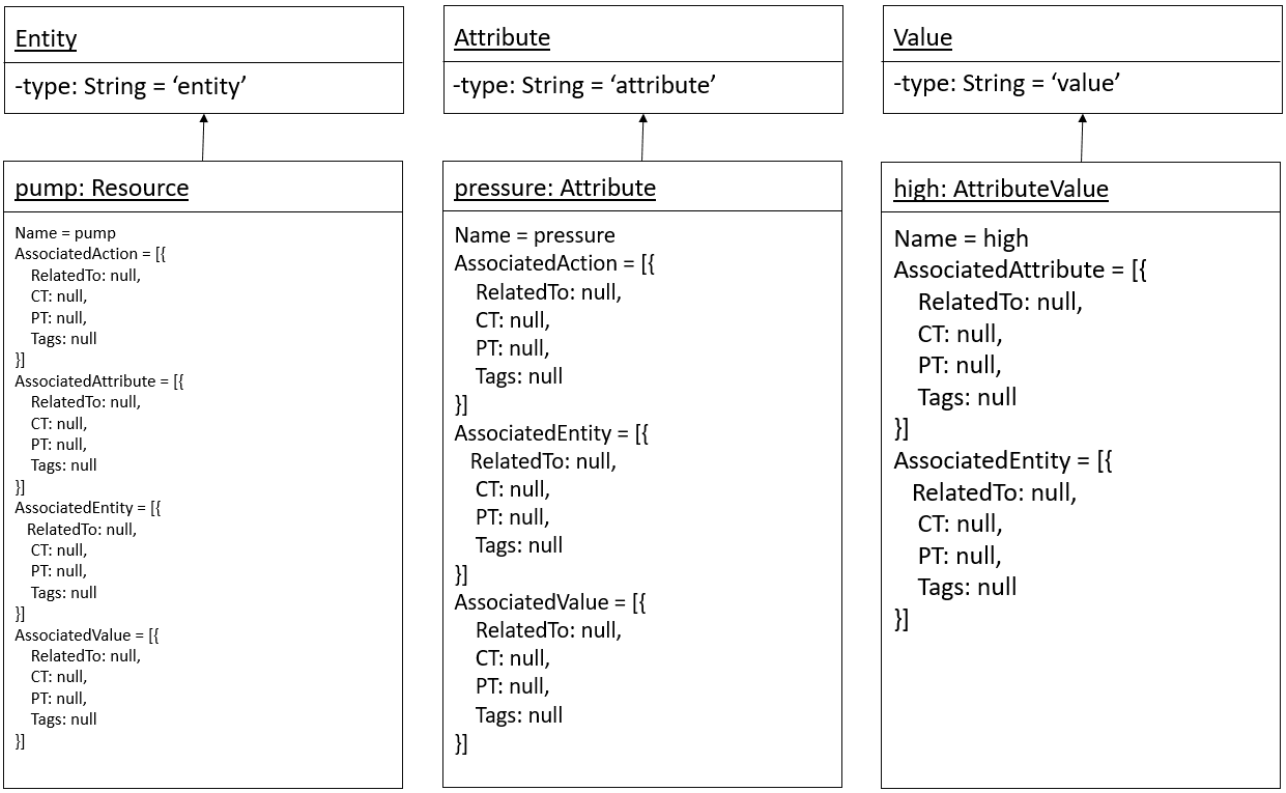


Figure 3.2: Three different nodes. Each representing a term, phrase or number and their classes, properties and sub-properties in the domain knowledge. Values are updated when new iText is read.

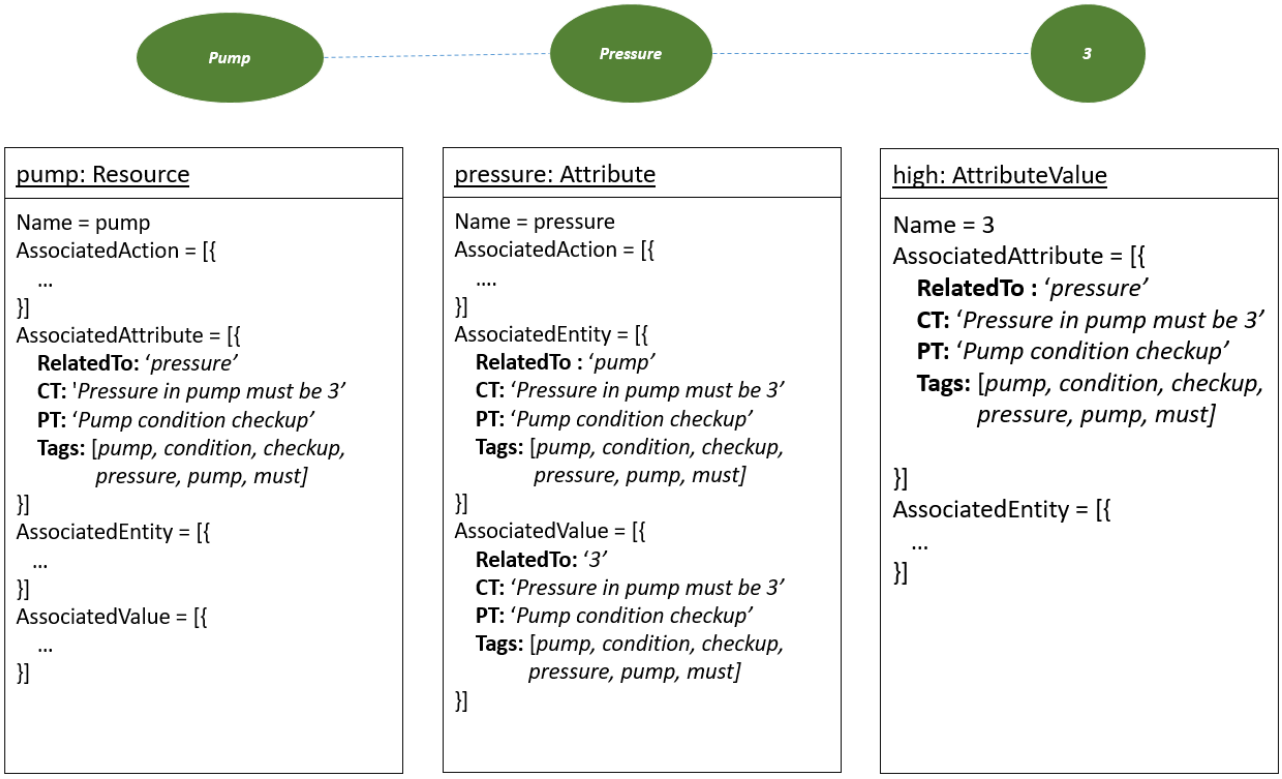


Figure 3.3: Three different nodes found in same iText and are connected with each other. Each representing a term, phrase or number and their classes, properties and sub-properties.

of these terms or phrases are assigned with properties and sub-properties of the class based on the category of the class. Hence, they can be considered as disconnected nodes. Once iText is learned, these nodes connect with one another using edges, based on the terms found in an iText, and that forms the HESN network. Figure 3.3 represent three terms related to each other, and their value gets updated when a particular iText is learned and when these terms were found in that particular iText. The update of these values and HESN has been explained in the latter part of the methodology section.

3.3 Human Experience Semantic Network (HESN)

HESN is the key component of our proposed knowledge base. Different terms and key phrases are identified from iTexts with the help of domain knowledge. There could be different operations or procedures in a document. Under each operation, there could be multiple instructions or procedure that talks about how to accomplish that particular operation. There could be the same term or key phrase in different operations. HESN represents the knowledge network that shows the association or relation of a term or key phrase with other terms or key phrases based on different operations. Each of these terms or key phrases could be an entity, action, attribute or value. The network is represented in the form of nodes and edges that constitute a

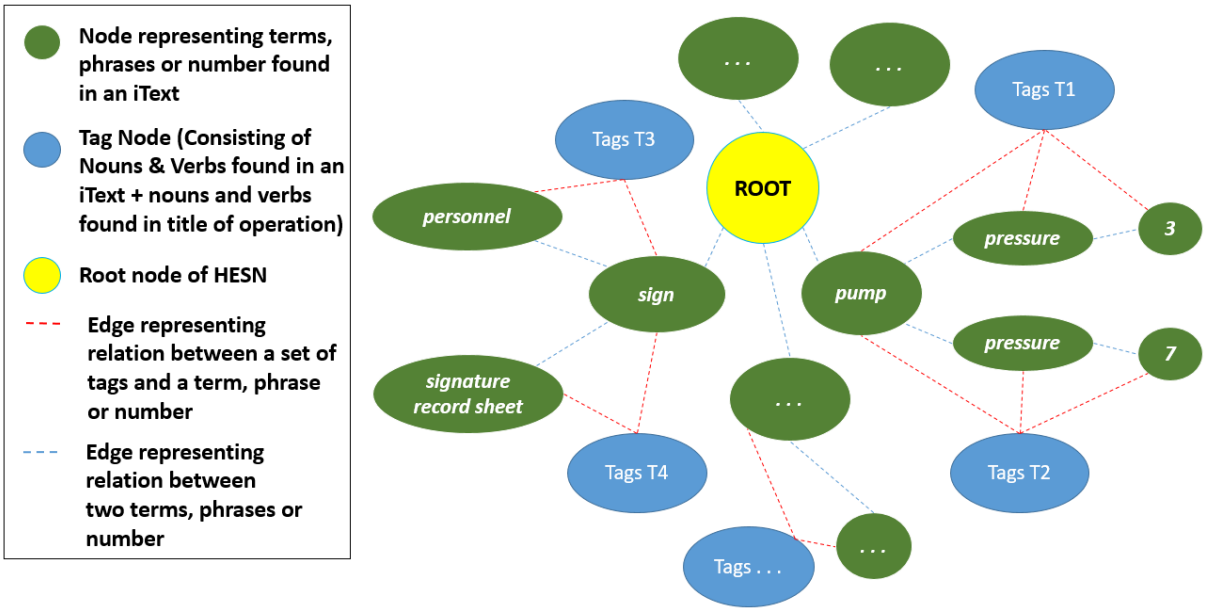


Figure 3.4: Human experience semantic network (HESN).

tree or undirected graph. Figure 3.4 represent a small glimpse of HESN where nodes and edges are connected. Figure 3.3 represents detailed information about each node where it is observed that the three terms 'pump', 'pressure' and '3' have relationship among one another based on a particular iText.

There could be six types of relationships based on the category of classes— i) entity-action (E-Ac), (ii) entity-entity (E-E), (iii) entity-attribute (E-Att), (iv) entity-value (E-V), (v) action-attribute (Ac-Att), and (vi) attribute-value (Att-V). The relationship is always created among two terms or key phrases. The properties of classes of each category are also designed based on these relationships. For example, the

properties of a class under the category 'attribute' are 'Name', 'AssociatedAction', 'AssociatedEntity' and 'AssociatedValue', as shown in Figure 3.3. The property 'AssociatedAction' retains information about another term that falls under the class of category 'Action'. Similarly, the properties 'AssociatedEntity' and 'AssociatedValue' retain information about other terms that fall under the class of category 'Entity' and 'Attribute Value' or 'Value', respectively. That is why it is only possible to relate an attribute term with only an entity, action or value. But not with other attribute terms. Similarly, an action term can only be related to another entity or attribute term, not with any other action term or value. If found, they will not be considered. HESN consists of different terms, represented as nodes, connected among each other by edges. Hence, an edge between two nodes, which could be a term, phrase or number, represent any one out of six mentioned relationships.

Our proposed HESN is considered a semantic network. A semantic network is a graphical representation of knowledge that can be used to do complex reasoning about knowledge [57]. As HESN retains the relations, semantics, and information about different terms and key phrases, and captures the knowledge and experience from iTexts existing in operational documents, thus it is called the human experience semantic network (HESN). The way semantics has been represented in HESN is with the help of classes of each term. Whenever there is a relation between two terms in

HESN, it is possible two know the meaning of the relationships since each of these terms is classified in the domain knowledge. And these classes help to know about the semantics of the term. For example, whenever there is a relation between the term 'engineer', classified as 'humanRole', and 'sign,' classified as 'action,' it is possible to infer that there is an action called 'sign' that has to be performed by a human called 'engineer.' HESN is also dynamic and adaptive in nature as the network expands when new iText is learned and values of the nodes update dynamically. Creating relations among different terms or key phrases based on the operation is performed with the help of tags. HESN also has qualitative and quantitative features. The methodology of creating relations among different terms and phrases and the use of tags is explained in the latter part of the research methodology section of this thesis.

3.4 Entity, Action, Attribute and Value Recognition and Linking

Domain knowledge is used to deal with recognizing terms and key phrases, which could be an entity, action term, attribute term, or some value. If named entities, action, attribute or any value, that consists of more than one word are identified, each word of that named entity, action, attribute or value is concatenated to make

it a single word. For example, water pump = waterpump. This helps in making the relationship among the words or key phrases easier later on. An attribute could be terms like pressure, height, condition, etc. Its value could be high, low, poor, etc. It could also be a numeric value. The domain knowledge consists of all these terms, except for the numeric values. Once the identification and concatenation are made, the next task is to establish relationships among the words or phrases or numbers.

At first, the stop words are removed from the sentence except for a few, which are "on," "in," "this," "have," "has," and "should." Afterwards, a grammar pattern-based linguistic matching is done with the help of a library named spaCy [1]. This helps to identify the direct dependency of a word over another word in a sentence in the form of a duplet. Each of these duplets is further processed and reorganized. Figure 3.5 shows the algorithm using which the tags are created and duplets are generated from each iText. Tags are the nouns and verbs found in an iText. A set of tags are used against each duplet. It helps to identify from which particular iText, the duplet was generated. Furthermore, this information helps to distinguish the relationships between different terms and key phrases based on different operations. The use of tags is explained further in the latter part of the research methodology section of the thesis.

From the algorithm, OP in Step 2 refers to a set of instructions having a title or

- Step 1: start
- Step 2: read ***T[i]*** from ***OP***
- Step 3: tokenize ***T[i]*** using ***NLP***
- Step 4: extract ***N*** and ***V*** from ***T[i]***
- Step 5: ***TAGS*** = [***N***, ***V***]
- Step 6: if ***T[i]*** = ***CT***,
  ALLTAGS = TAGS + PTAGS, Go to Step 8
- Step 7: else if ***T[i]*** = ***PT***, ***PTAGS = TAGS***, Go to Step 16
- Step 8: ***sp*** = spacy.load('en_core_web_lg')
- Step 9: ***T[i]*** = **removeStopWords(*T[i]*)**
- Step 10: ***doc*** = ***sp(T[i])***
- Step 11: ***DD*** = **getAllDD*(doc)***
- Step 12: for ***d*** in ***DD*** –
  if (***d[0]*** or ***d[1]*** does not contain ***DK*** or numeric value)***,*** remove ***d*** from ***DD***
- Step 13: for ***d*** in ***DD*** –
  if (***d[0]*** does not contain ***DK*** or numeric value),
  for ***w*** in ***DD*** –
  if (***d[0]*** in ***w*** and ***d[1]*** not in ***w***)
  if ***d[0]*** = ***w[0]***, ***d[0]*** = ***w[1]***
  else ***d[0]*** = ***w[0]***
  break
  else if (***d[1]*** does not contain ***DK*** or numeric value),
  for ***w*** in ***DD*** –
  if (***d[1]*** in ***w*** and ***d[0]*** not in ***w***)
  if ***d[1]*** = ***w[0]***, ***d[1]*** = ***w[1]***
  else ***d[1]*** = ***w[0]***
  break
- Step 14: for ***d*** in ***DD*** –
  if (***d[0]*** and ***d[1]*** does contain ***DK*** or numeric value), remove ***d*** from ***DD***
- Step 15: **updateHESN*(DD, ALLTAGS)***
- Step 16: ***i*** = ***i*** +1
- Step 17: if ***T[i]*** exist, go to step 2
- Step 18: else stop

Figure 3.5: Algorithm of creating tags and duplet formation

operation name (PT) and one or more instructions (CT). T[i] represents each iText which could be a PT or CT. 'N' and 'V' in the algorithm means all nouns and verbs extracted from that particular iText. 'TAGS' in Step 5 denote all the Nouns and Verbs of T[i], whereas "ALLTAGS" in Step 6 denote tags of that particular iText and the 'PTAGS'. 'PTAGS' are the tags extracted from PT. All necessary components of the spaCy library is loaded and assigned to 'sp' in Step 8. It can now be used to perform tasks like finding word dependencies from within a sentence. In Step 9, the stop words are removed from the iText except for a few, which are 'on', 'in', 'this', 'have', 'has', and 'should'. In Step 10, the iText is processed using 'sp' to get valuable insight, such as direct word dependencies, parts of speech tag for each word, etc. In Step 11, the function "getAllDD" returns word dependency for each word in the sentence in the form of duplets. Each element in the duplet is represented as d[0] and d[1], as shown in Step 12. "DK" consists of all terms found in domain knowledge. Each duplet consisting of two elements are processed and if any one of the elements of duplet does not consist of terms or phrases from DK or if it is not a number, that entire duplet is removed from DD. In Step 13, relation between two terms, phrases, that are classified in DK, or number has been established. The final "DD" found in Step 14, after ending the loop, consists of the sorted duplets. Concatenation of the duplets creates a small network for that particular iText, as shown in Figure

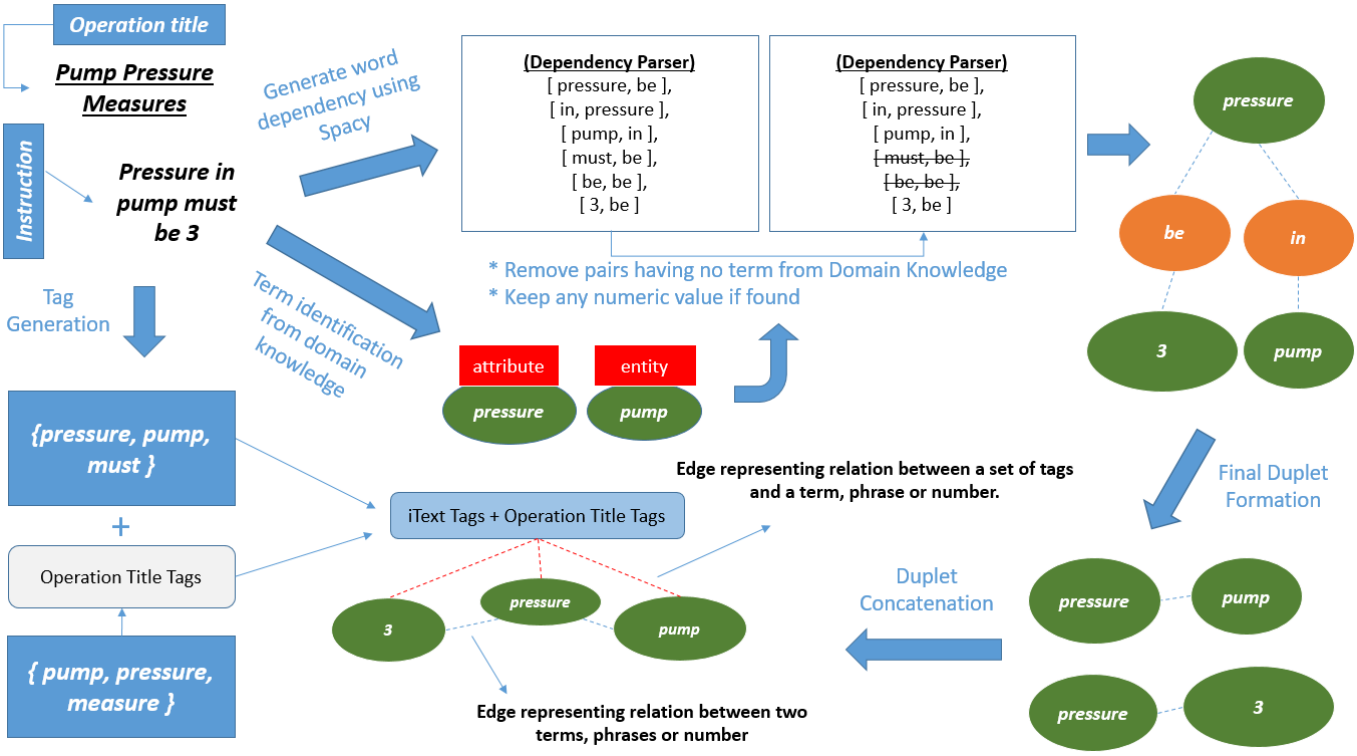


Figure 3.6: Generation of duplets and formation of small network from iText.

3.6. This network is the building block of HESN. Figure 3.6 visually represents the methodology of how a small HESN network is generated from an iText. Step 15 is described in section "Update HESN" of this thesis.

3.5 Tag Generation and Relation Tracking

When it comes to iTexts, it is essential to track the information about different terms and phrases provided in different sets of instructions or operations. If we again

consider Figure 3.6, we get the entity here as “pump”, and its attribute is “pressure”. The value is mentioned as 3. Let us consider this value for ’pump’ for operation OP1. There could be another operation OP2 where the entity and attribute are the same, but the value is 7. In this case, two different values are obtained having the same attribute of the entity but for different operations, OP1 and OP2. In order to keep track of this knowledge, tag plays an important role. Figure 3.7 shows how relations of the same entity are structured for two different operations. Tags are termed in this research as the nouns and verbs extracted from text, having word’s character length greater than 2 for verbs and any character length for nouns. For every network that is generated from each instruction, tags are added against them. These tags contain the nouns and verbs extracted from that particular instruction and the title of the operation under which the instruction is situated. Considering the same example from Figure 3.6, if T1 is considered as the set of tags for those associations found in the small network, formed from that particular iText, then T1 consists of the nouns and verbs of that iText (CT), along with tags generated from the title of its operation (PT). This takes place for every instruction under the same operation. This helps to keep track of which information is coming from which operation. The process of extracting nouns and verbs from text is done with the help of a popular natural language processing technique called Parts-of-speech (POS) Tagging. It has been

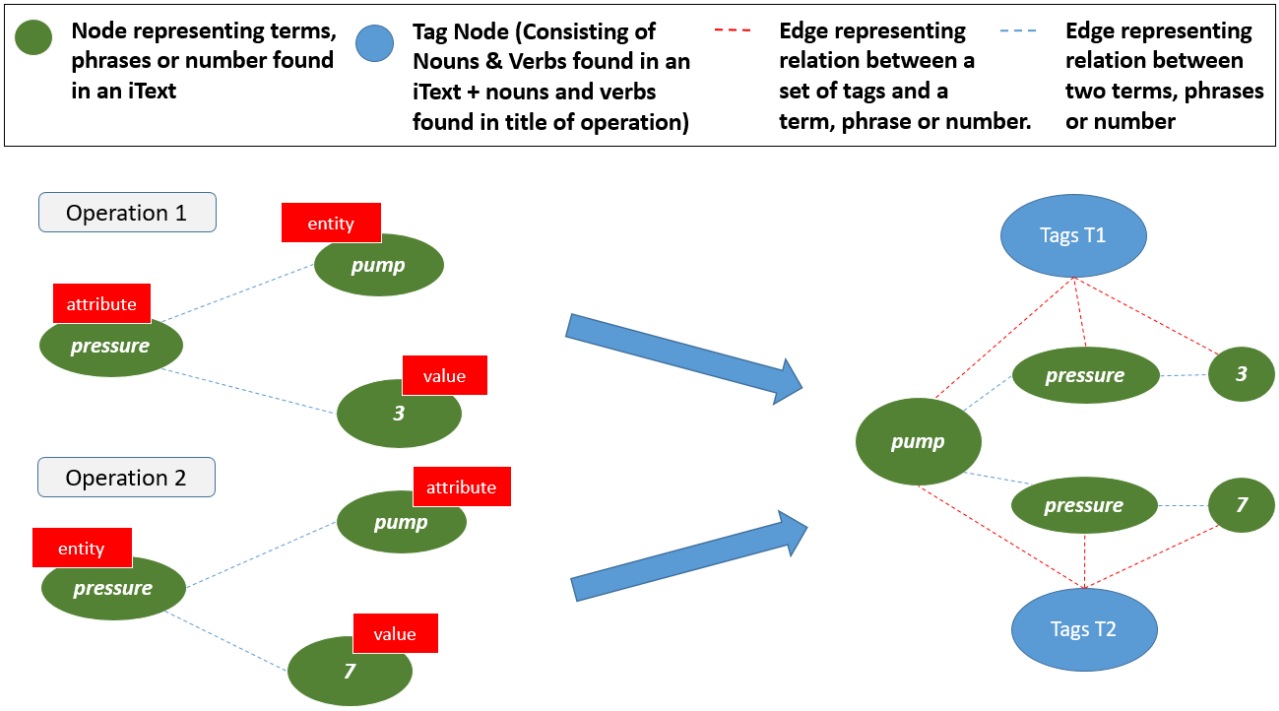


Figure 3.7: Updating value of same entity from two different iText for two different operation which shows how HESN is updated

used in this research for generating the tags from each iText which could be PT or CT. The process is shown in Figure 3.8. The text 'Signing the signature record sheet is mandatory for all personnel' is processed using the POS Tagging technique, which marks each word in a corpus to a corresponding part of a speech tag. This helps to identify which terms are nouns and which ones are verbs, as only nouns and verbs are needed to generate tags. As a result, from the sentence shown in Figure 3.8, the final generated tags are signature, personnel, record, sheet and sign.

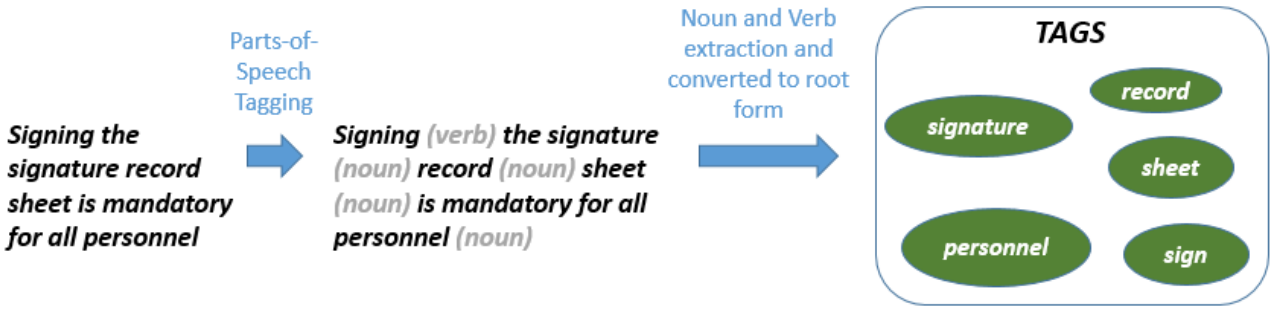


Figure 3.8: Extracting nouns and verbs from text with the help of POS Tagging technique

3.6 Update HESN

From Figure 3.6, it is observed how a small network is generated from each iText consisting of the relationship among terms of entity, action, attribute, and its value and how the respective tags are generated from that particular instruction (CT) and title of the operation (PT). HESN consists of nodes and edges. Figure 3.3 represent detailed information about a node. Whenever a new relation is created between two terms or key phrases, the property of both of the terms is updated. For example, in Figure 3.3, the term 'pump', 'pressure' and '3' are related and was found from the iText 'Pressure in pump must be 3', which is a CT and it is an instruction under the PT or operation title 'Pump condition checkup'. The term 'pump' belongs to the class 'resource', which is under the category of entity. The term 'Pressure' is an attribute.

The property 'AssociatedAttribute' of 'pump' is updated with the term 'pressure', which indicates that an entity-attribute (E-Att) relationship has been established from this particular iText between the entity 'pump' and the attribute 'pressure'. Similarly, the 'AssociatedEntity' property of the term 'pressure' is updated with the information of 'pump'. An attribute-value (Att-V) relationship is also established between the term 'pressure' and '3'. Since 3 is a number and belongs to the class value, as all numbers are identified as an attribute value or value category, hence the 'AssociatedValue' property of the term 'pressure' is updated with the value 3. The 'AssociatedAttribute' property of the value 3 is updated with the term 'pressure'. All three terms share the same sub-properties CT, PT and Tags since they are found in the same iText under the same operation name.

Two types of relationships were observed, one is entity-attribute (E-Att), and the other one is attribute-value (Att-V). Since these two types of relationships fall within the six types of relationships mentioned earlier, as a result, the relationship between these terms is considered valid. They form a small part of the HESN network. In this way, values are updated dynamically, and relationships are established among different terms when new iTexts are learned. From the HESN, if a node is considered, for e.g. 'pump', it is possible to know what other terms are associated with 'pump' and in which iText the relationship was established. HESN not only create relationship

among different terms but also provide an idea from which particular iText and operation the relationship was established. This is a unique feature of our proposed knowledge base consisting of HESN and domain knowledge. As the information of HESN is stored in the knowledge base, the knowledge base is also updated when HESN is updated.

3.7 Technology Used for Implementation

The entire application is a web-based application and is divided into three independent applications. The first one is the User Interface. It is built with a front-end JavaScript framework called Vue.js. It supports modern front-end design and development features. It is run using Node.js, which is a JavaScript run time environment. The second application is developed using Python, which is used to read and learn from iText found in documents. Python provides interesting libraries which help to extract texts from pdf. Flask, a popular library, is used in Python to develop the Application Programming Interface (API). Transfer of data takes place with the help of API. Spacy, another library, is also used in Python for detecting the word-dependency in iTexts. The third application is the main back-end application, built with Node.js, which is used to communicate between User Interface and Database with the help of API. The APIs are developed in the back-end application with the help of a Javascript

framework called Express.js. This back-end data application helps to transfer data to and from the database and user interface. The database that has been used is MySQL. It is a relational database.

The reason behind creating three separate applications is to make each application independent of the other in terms of development and deployment. This agile process helps in easily expanding the application. Reporting any software bugs or errors also becomes more manageable.

Chapter 4

Case Study

This chapter presents a case study to understand the methodology of learning and capturing knowledge from iText. The case study will help to know how entity, action, attribute and attribute values are identified. Moreover, it will help to understand how HESN is formed and updated. Furthermore, it will demonstrate the entity, action, attribute and value recognition technique, tag generation approach, linking terms against each operation and HESN formation. Finally, a few queries are used to show the possible reasoning that can be done from our proposed knowledge base and its overall impact with be discussed.

4.1 Description of the case Study

In this case study, a set of iText has been demonstrated. It consists of a title and a few instructions or procedures that talk about accomplishing the task or operation. The case study will help to understand how our proposed methodology works. It will show how the domain knowledge is used to recognize different terms and phrases and how HESN is formed and updated. With the help of the example provided in Figure 4.1, it is possible to know, step by step, how knowledge is captured from iText.

4.2 Reading and Learning from a set of iText

Let us consider a few iTexts. It has a title, which is the parent text (PT). Next, it has a few instructions, which are the child text (CT). In figure 4.1, a set of iText is represented consisting of PT and two CTs. In this case study, this set of iText will be processed, knowledge will be captured, and HESN will be formed or updated in a way that will help to understand our proposed methodology.

4.2.1 Entity, Action, Attribute and Value Recognition

Firstly, different terms from the iText are detected based on domain knowledge. This is done for the instructions or CTs only. From the first CT, the terms 'personnel',

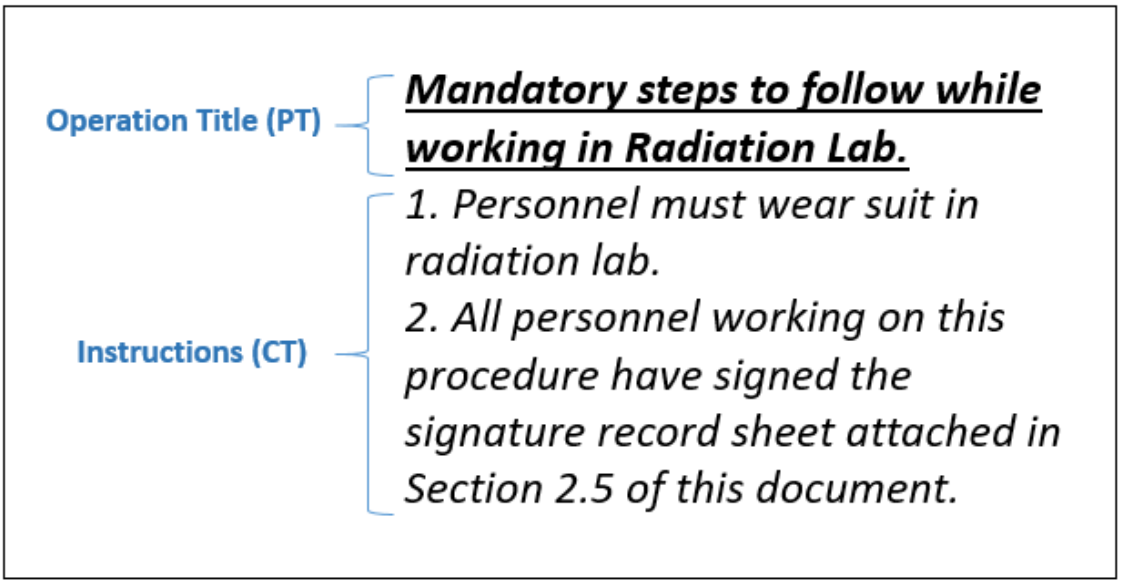


Figure 4.1: iTexts consisting of Operation Title (Parent Text or PT) and instructions (Child Text or CT)

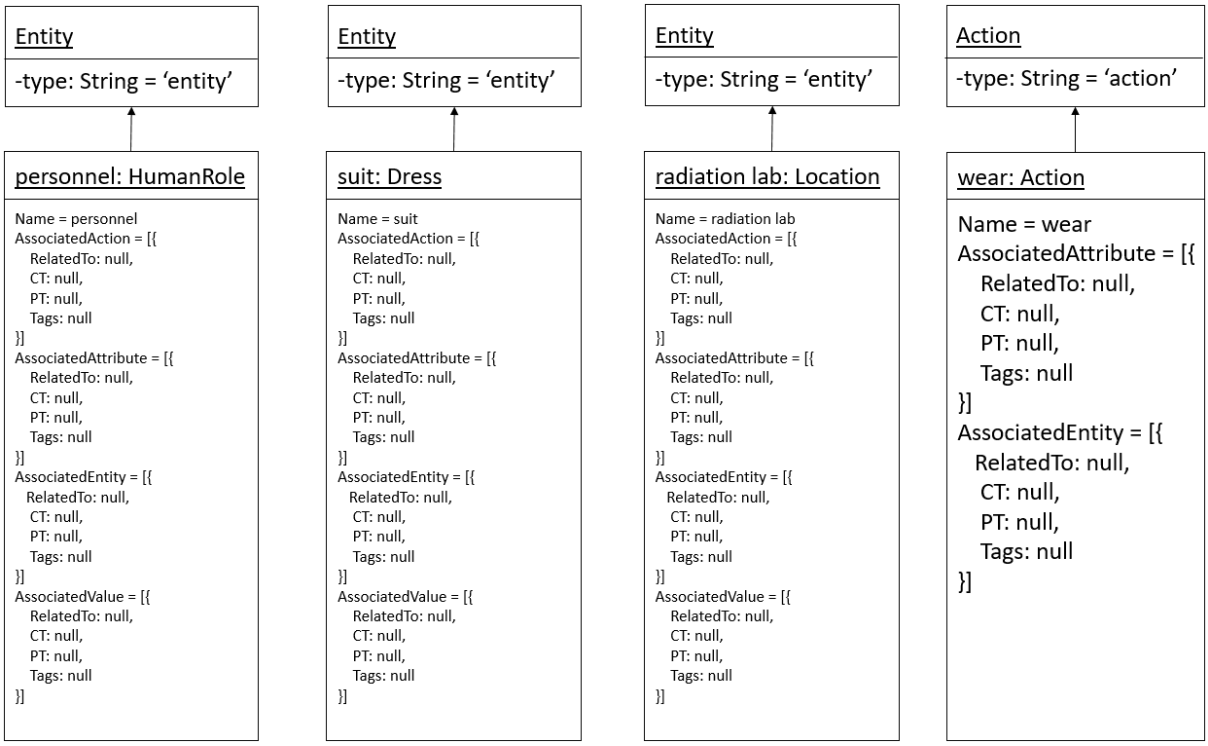


Figure 4.2: The four terms or phrases, detected in the first CT of figure 4.1, classified by a domain expert in the domain knowledge and their initial parameters and values before learning the CT

'wear', 'suit' and 'radiation lab' are detected as 'entity', 'action', 'entity' and 'entity' respectively. This is done with the help of domain knowledge, which already consists of these terms. Figure 4.2 shows how these four terms or phrases are initially classified by a domain expert in the knowledge base before learning from the first CT. These could be considered as individual nodes which are not connected with any other nodes representing terms or phrases yet. Similarly, from the second instruction or CT, the

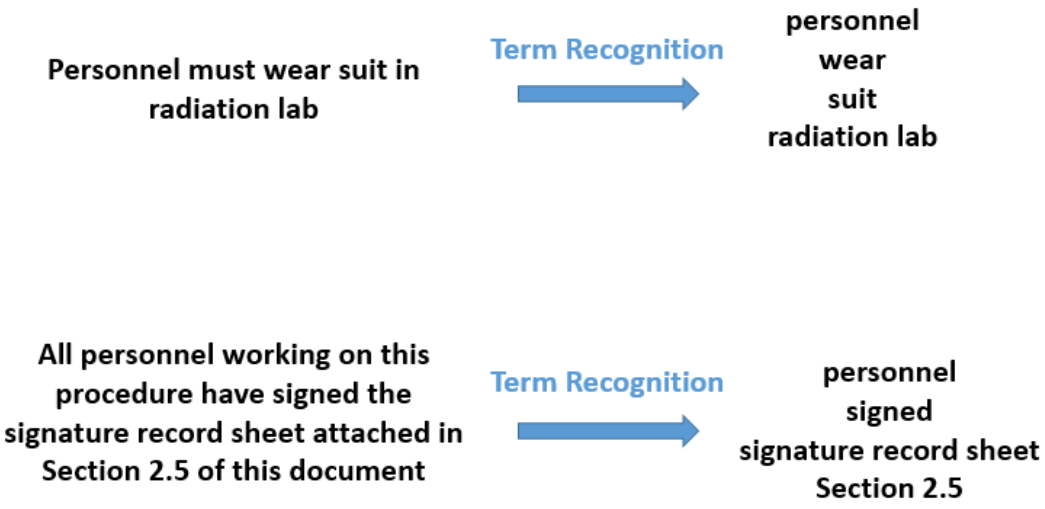


Figure 4.3: Term Recognition based on Domain Knowledge

terms 'personnel', 'signed', 'signature record sheet' and 'section 2.5' are detected as 'entity', 'action', 'entity' and 'entity' with the help of the domain knowledge. Figure 4.3 shows the recognition of terms from the first CT (CT1) and second CT (CT2) from the set of iTexts as shown in Figure 4.1.

4.2.2 Tag Generation from each iText

As explained earlier, tags are nothing but the nouns and verbs found in a text. A rule is followed here which goes like this -

1. Verbs are extracted as tags when its word's character length is greater than 2.

2. Nouns are extracted as tags having any word's character length.

With the help of a Natural Language Processing technique called "Parts-of-Speech Tagging," each iText, PT and CTs, are processed to get the nouns and verbs from each one of them. From the PT "Mandatory steps to follow while working in Radiation Lab", the extracted tags are "steps" "follow", "working", "radiation" and "lab". Similarly, from the CT "Personnel must wear suit in radiation lab", the extracted tags are "personnel", "must", "wear", "suit", "radiation" and "lab". And from the CT "All personnel working on this procedure have signed the signature record sheet attached in Section 2.5 of this document", the extracted tags are "personnel", "work", "procedure", "have", "sign", "signature", "record", "sheet", "attach", "section" and "document". The process is shown in figure 4.4 and figure 4.5. These tags are linked with each relationship among the identified terms of the respective iText. According to the described tagging approach explained earlier in this research, if all tags of the relationships among terms found in CT1 are considered as ACT1, then ACT1 = Tags of CT1 + Tags of PT. Hence, ACT1 = {step, follow, work, radiation, lab, personnel, must, wear, suit, radiation, lab}. Similarly, if all tags of the relationships among terms found in CT2 is considered as ACT2, then ACT2 = {step, follow, work, radiation, lab, personnel, work, procedure, have, sign, signature, record, sheet, attach, section, document}.

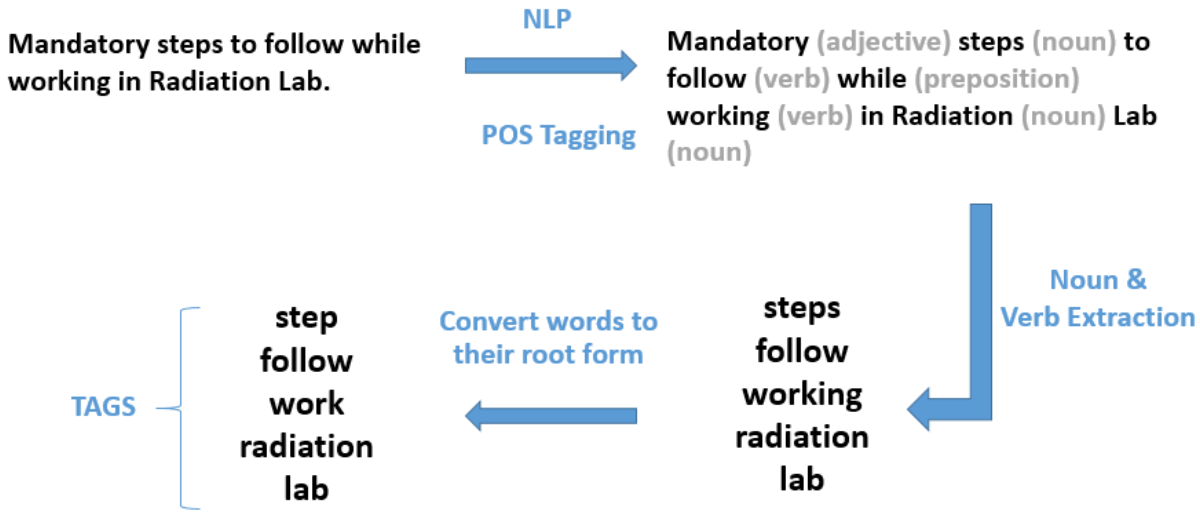


Figure 4.4: Process of generating tags from iText and tags extracted from PT of example shown in figure 4.1

4.2.3 Linking Terms and HESN formation

The next step is to link the identified terms. Before that, each identified term is processed to convert them into one word for the terms or phrases that have more than one word. For e.g. the phrase "radiation lab" is converted to "radiationlab". Similarly, the phrase "signature record sheet" is converted to "signaturerecordsheet". The process is followed for every term identified from each CT. Afterwards, the word dependency is calculated to know which word is dependent upon which other words. From this process, duplets are formed from every two terms that are dependent on each other. These duplets are processed to get the final duplets. The algorithm

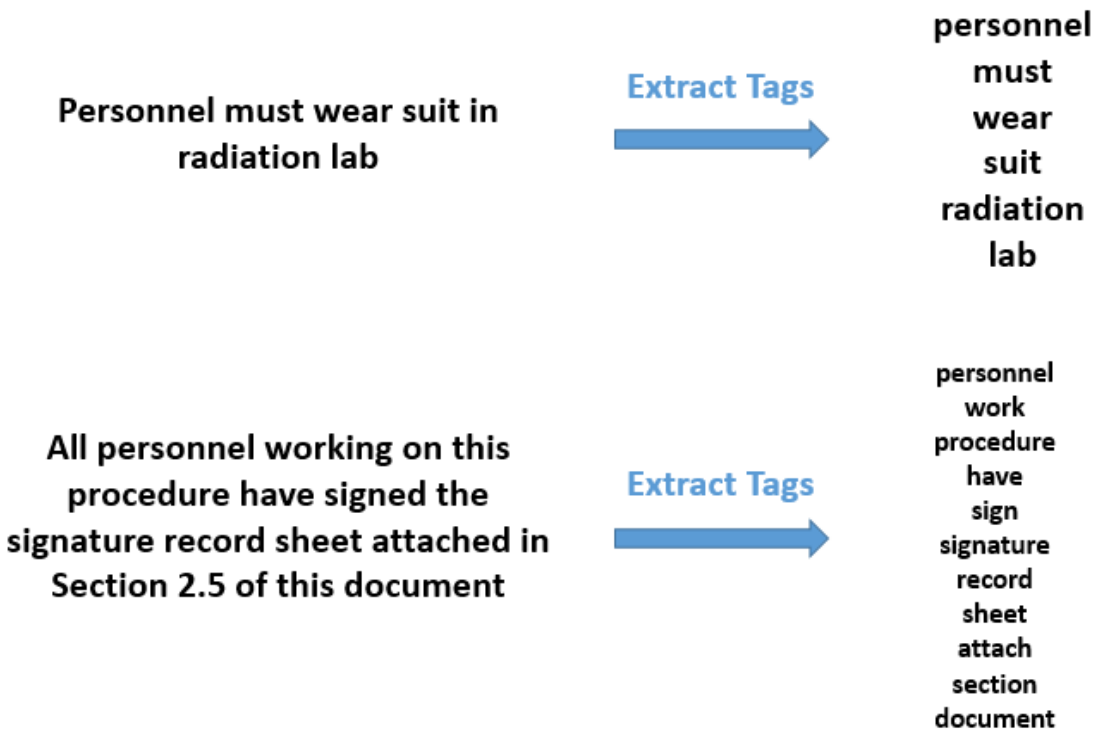


Figure 4.5: Tags extracted from CT1 and CT2 of the example shown in figure 4.1

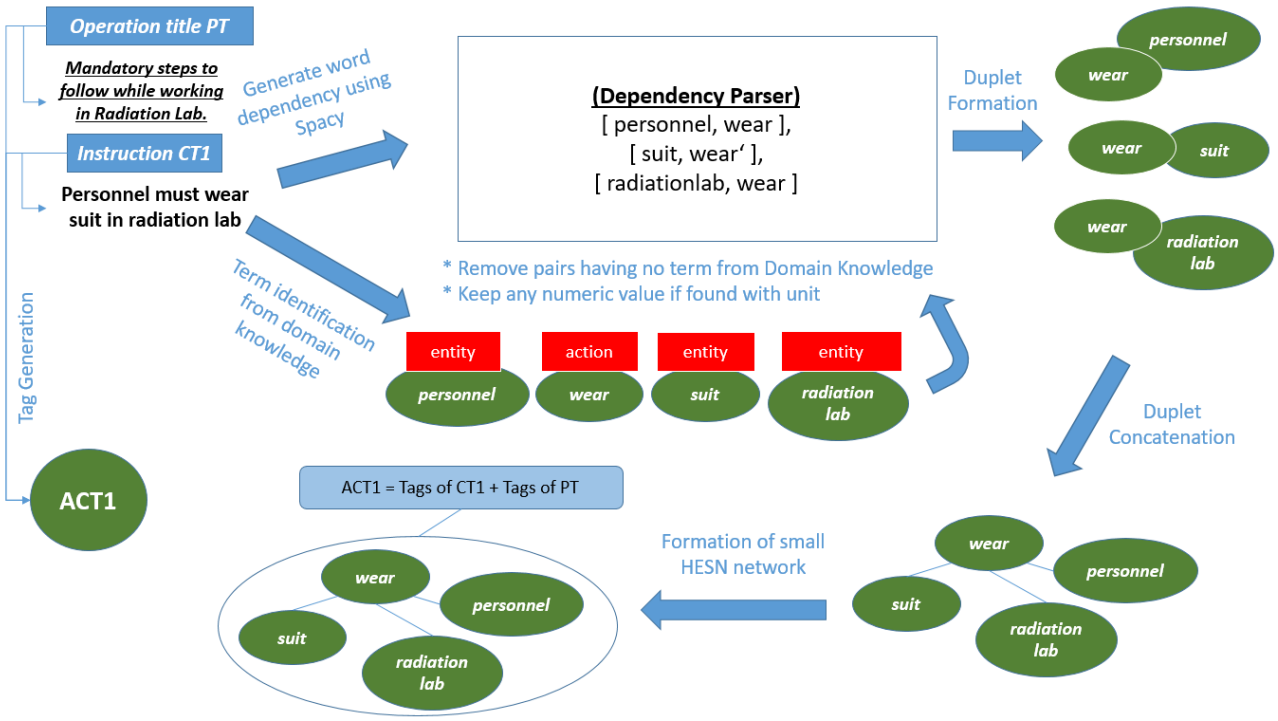


Figure 4.6: Term Linking and HESN formation for CT1

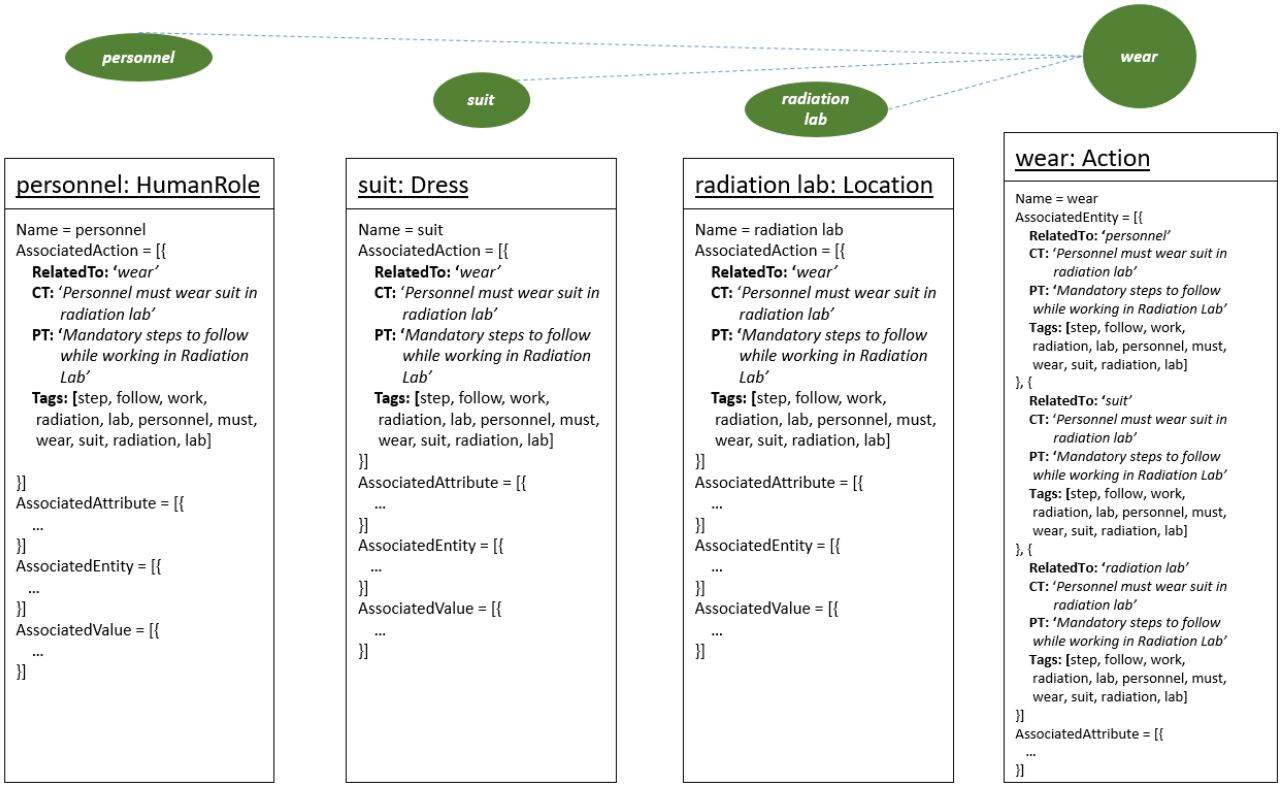


Figure 4.7: The four terms or phrases, detected in CT1 of figure 4.1, classified by a domain expert in the domain knowledge and their parameters and values updated after learning from CT1

that does the work is shown in Figure 3.5 and has already been described earlier. Finally, all these duplets together form a small network. This network is the building block of HESN. This small network is added to the existing HESN network. Thus, the knowledge base is updated as well. The entire process is shown in figure 4.6 for the example shown in figure 4.1. The process of HESN formation shown in figure 4.6 is for CT1 only. All the processed duplets are concatenated to form the desired network. This small knowledge network formed from CT1 is finally tagged with ACT1. ACT1 consists of all the tags of CT1 and all the tags of PT. This tagging helps to know which information is coming from which iText. Figure 4.2 shows how the terms 'personnel', 'wear', 'suit' and 'radiation lab' look like in the domain knowledge before learning any iText. Figure 4.7 represents the same four terms after learning from iText CT1 and how their values updated after learning. If the term 'personnel' is considered, it is associated with the term 'wear', and there is an Entity-Action (E-Ac) relationship between them. Since this relationship is identified as one of the six types of relationships as mentioned earlier, so the relationship is valid. As a result, the 'AssociatedAction' property is updated with the term 'wear'. The same kind of relationship is found between the terms 'suit' and 'wear', and 'radiation lab' and 'wear'. As a result, the 'AssociatedAction' property is updated for the terms 'suit' and 'radiation lab' in the same way as that for 'personnel'. If

the term 'wear' is considered, it is associated with three other entities in CT1. So, in the 'AssociatedEntity' property, the information about all other three terms got updated, and it is also clear that in which iText these relationships were established. All these four terms share the same PT, CT and Tags since they were found in the same iText CT1 under the same operation PT.

Now, if a new iText is read from a completely different operation and an Entity-Action (E-Ac) relationship is established between the term 'wear' and the new term 'spectacle', which is already defined in the domain knowledge, then both of their values will update in the same way. If figure 4.8 is observed, the 'AssociatedEntity' property of the term 'wear' is added with information about the term 'spectacle'. This time, it has different CT, PT and Tags, and that represents where the relationship was established between the term 'wear' and 'spectacle'. The 'AssociatedAction' property of the term 'spectacle' is also updated with the information of the term 'wear', and in this case, they share the same CT, PT and Tags since the relationship was established in the same iText.

Coming back to the main example, for CT2, the small HESN network is shown in figure 4.9

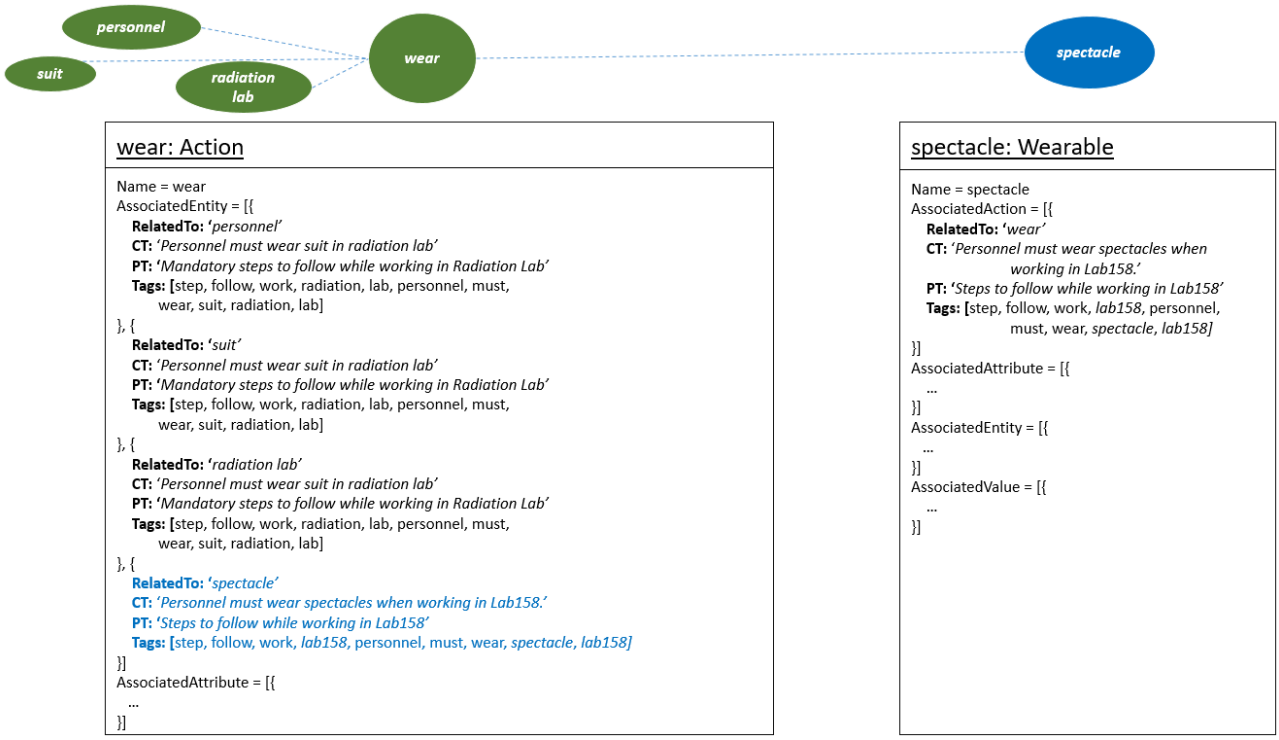


Figure 4.8: Relationship established between the term 'wear' and 'spectacle' and their properties updated accordingly.

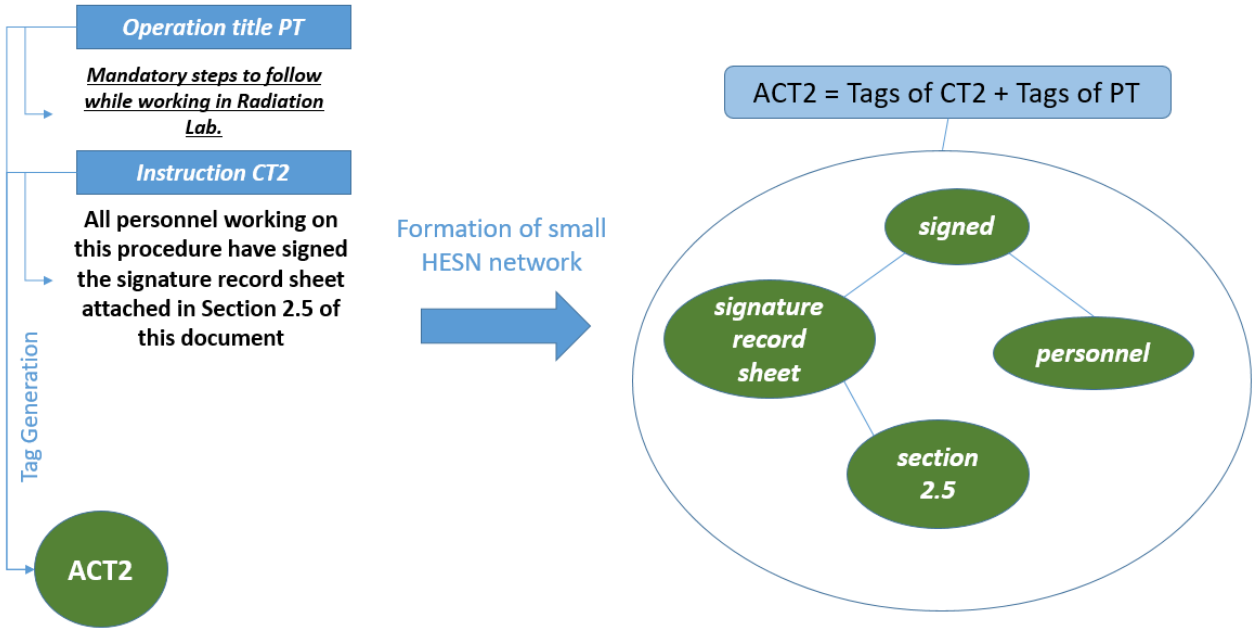


Figure 4.9: Term Linking and HESN formation for CT2

4.2.4 Adding Another Operation to HESN

Let us consider another set of iTexts. It has a title, which is the parent text (PT 2) and the child texts CT2-1 and CT2-2. In figure 4.10, a set of iText is represented consisting of PT 2 and two CTs. Figure 4.11 shows how each of the child texts is processed and how HESN is formed from each one of them. Their properties and values are updated in the same way as seen earlier in the previous example. The examples in figure 4.1 and 4.10, both are little bit similar to one another. However, the difference could be made with the help of the tags. Furthermore, this helps reasoning based on different parameters that are captured from the tags.

4.3 Possible Reasoning and Impact of the Knowledge Base

Although this thesis does not propose any information retrieval mechanism, a tag matching approach can help retrieve information from the knowledge base based on different parameters and values. Let us consider the following two queries -

1. What should all personnel wear in the Radiation Lab?

2. What should all personnel wear in Red Zone?

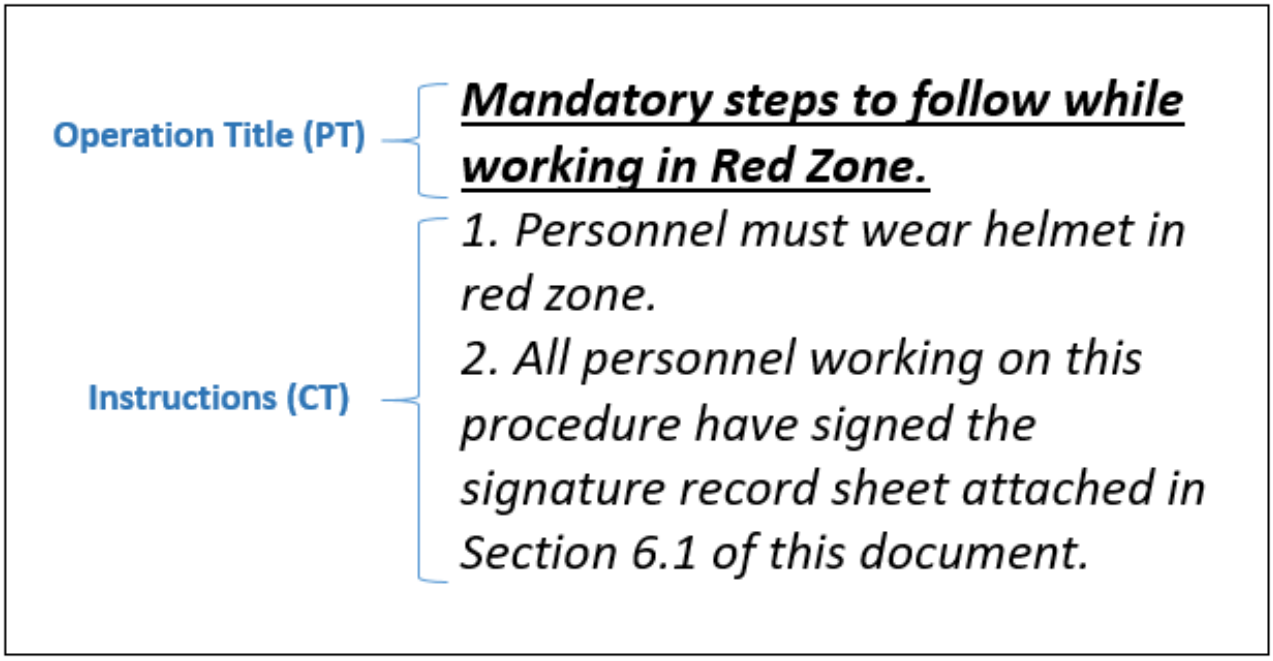


Figure 4.10: iText consisting of details of additional operation

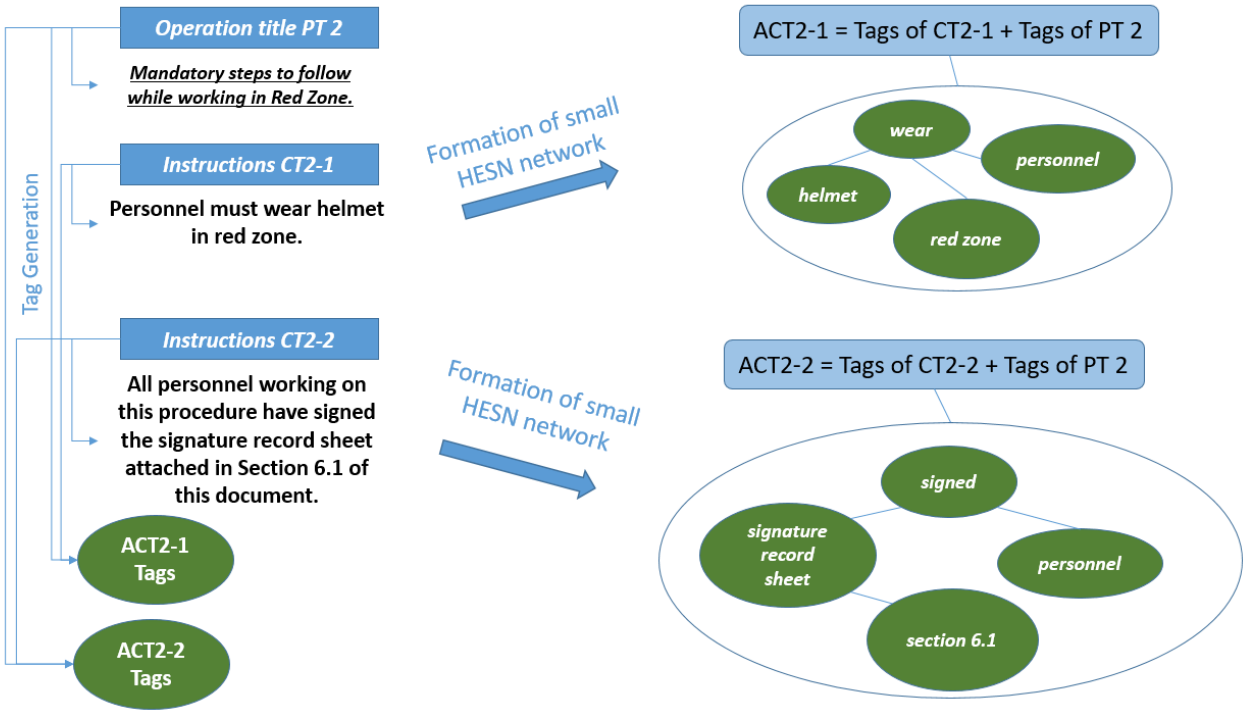


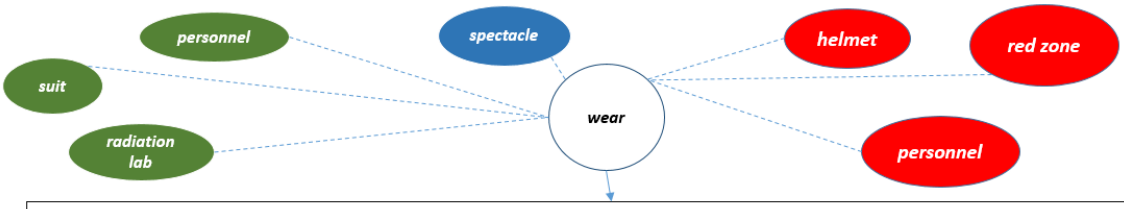
Figure 4.11: Term Linking and HESN formation for CT2

From the first query, the tags could be extracted using the POS Tagging process of the Natural Language Processing technique. Hence we get the tags 'should,' 'personnel,' 'wear,' 'radiation,' 'lab.' In our knowledge base, tags exist against each set of relationships. Now, these tags from the first query match the most with the tags of CT1 of operation one shown in figure 4.1. Similarly, the tags found from the second query are 'should,' 'personnel,' 'wear,' 'red,' 'zone.' And these tags match the maximum with the tags of CT2-1 of operation two shown in figure 4.10. In this way, with the help of the tag matching approach, the reasoning is possible from our proposed knowledge base.

Moreover, if the term 'wear' is considered, it is a node, and so far, it has been observed that it is related to other terms, namely 'personnel', 'suit', 'radiation lab', 'red zone', 'helmet' and 'spectacle' as shown in figure 4.12, which is a small HESN network. In the figure, the same color of the nodes represent that they are from the same iText. For e.g. the red nodes 'helmet', 'red zone' and 'personnel' established a relationship with the node 'wear' in the same CT 'Personnel must wear helmet in red zone' under the same PT 'Mandatory steps to follow while working in Red Zone'. That is why the red texts in 'AssociatedEntity' property of the term 'wear' represent information about those red nodes. Similarly, the green texts and the blue text in 'AssociatedEntity' property of the term 'wear' represent information about

those green nodes and blue node, respectively, as shown in figure 4.12. Similar color means that the relationship was established in the same iText or CT under the same PT.

From this single term 'wear', it is possible to know in which other operations this particular action took place and with what other nodes. The term 'suit' is classified as 'dress', and it falls under the category of Entity or Name phrase. The class 'dress' defines the term 'suit'. As a result, it is possible to know in which operations this dress is required to wear, who should wear this dress and where. By 'who', it could be a term that belongs to the class 'humanRole' as, under this class, the domain expert defines different human roles such as personnel, engineer, manager etc. By 'where', it could be a term that belongs to the class 'location' where terms like 'radiation lab', 'red zone' etc., are defined by the domain expert. All these complex reasoning is possible from the proposed knowledge base. HESN captures all these experiences in the form of nodes and edges. This network is powerful to capture knowledge from iText with the help of domain knowledge. Furthermore, the design of domain knowledge is also not complex. It is mostly assigning different classes to different categories and assigning different terms under each class. The properties, except 'Name', and sub-properties of the class are always fixed based on the category of class which is not required to be defined by the domain expert. The values are



wear: Action

Name = wear
AssociatedEntity = [
{ RelatedTo: '*personnel***' CT: '***Personnel must wear suit in radiation lab***' PT: '***Mandatory steps to follow while working in Radiation Lab***' Tags:** [step, follow, work, radiation, lab, personnel, must, wear, suit, radiation, lab] },

{ RelatedTo: '*suit***' CT: '***Personnel must wear suit in radiation lab***' PT: '***Mandatory steps to follow while working in Radiation Lab***' Tags:** [step, follow, work, radiation, lab, personnel, must, wear, suit, radiation, lab] },

{ RelatedTo: '*radiation lab***' CT: '***Personnel must wear suit in radiation lab***' PT: '***Mandatory steps to follow while working in Radiation Lab***' Tags:** [step, follow, work, radiation, lab, personnel, must, wear, suit, radiation, lab] },

{ RelatedTo: '*spectacle***' CT: '***Personnel must wear spectacles when working in Lab158***' PT: '***Steps to follow while working in Lab158***' Tags:** [step, follow, work, *lab158*, personnel, must, wear, *spectacle*, *lab158*] },

{ RelatedTo: '*personnel***' CT: '***Personnel must wear helmet in red zone***' PT: '***Mandatory steps to follow while working in Red Zone***' Tags:** [step, follow, work, red, zone, personnel, must, wear, helmet, red, zone] },

{ RelatedTo: '*helmet***' CT: '***Personnel must wear helmet in red zone***' PT: '***Mandatory steps to follow while working in Red Zone***' Tags:** [step, follow, work, red, zone, personnel, must, wear, helmet, red, zone] },

{ RelatedTo: '*red zone***' CT: '***Personnel must wear helmet in red zone***' PT: '***Mandatory steps to follow while working in Red Zone***' Tags:** [step, follow, work, red, zone, personnel, must, wear, helmet, red, zone] },
]
AssociatedAttribute = []

Figure 4.12: Relationship of the node or term 'wear' with other terms found in different iTexts. Similar color in nodes and similar color in the properties are used to represent that those nodes are from the same iText

updated when each iText is learned with the help of the algorithm described in figure 3.5. And that forms the HESN.

Chapter 5

Results and Validations

As discussed earlier, the textual pattern of regular text or paragraph is different from that of instructive text (iText). The structuring of knowledge from iTexts and capturing the human experience from it is not only about extracting or establishing relationships among the entities found in each iText. It is also about linking that information and relationships with different operations or operation titles. And this is done using tags, as shown previously in the Chapter 3 of this thesis. This helps to know what are the entities, actions and attributes that are existing in each iText. It also helps to know about the list of entities, actions and attributes for a particular operation. Every duplet consisting of two elements has a relation. The six types of relations between any two elements of a duplet may have are—(i) entity-action

(E-Ac), (ii) entity-entity (E-E), (iii) entity-attribute (E-Att), (iv) entity-value (E-V), (v) action-attribute (Ac-Att), and (vi) attribute-value (Att-V). Any other types of relationships found in iTexts are excluded and not considered for further analysis. They are just simply neglected.

5.1 Relation Extraction

The accuracy of relation extraction is measured based on the procedural and operational test documents provided by OPG. In total, 25 different types of sentences or iTexts were selected, and 102 relations were extracted. Each relation is made between 2 keywords or phrases or number. A total of 16 relations were ignored as they do not fall into previously mentioned six types of relations. That means, whenever a relation between an action term and another action term, for example, is found, that relation is ignored as this type of relation falls out of the scope of HESN relationships that we proposed. 79 relations were correctly extracted. Figure 5.1, 5.2, 5.3 and 5.4 represent 4 tables that show what duplets are generated from each iText. These results are shown to provide an example of how duplets are generated and finalized from each iText. Each of these duplets contains a relation, and the terms are already classified in the domain knowledge. In the "relation" column, "TRUE" means that a particular duplet follows one of the six types of relations that were previously mentioned, and

iText	duplets	relation	wrong	correct/total
personnel must wear suit in radiation lab	personnel (E), wear (Ac)	TRUE	0	3/3
	suit (E), wear (Ac)	TRUE		
	radiation lab (E), wear (Ac)	TRUE		
prerequisites have been completed	Prerequisites (E), completed (Ac)	TRUE	0	1/1
record channel number, date, start time and repositioning required in appendix B datasheet 1	record (Ac), channel number (Att)	TRUE	0	3/3
	date(Att), channel number (Att)	IGNORED		
	start time (Att), repositioning (Att)	IGNORED		
	repositioning (Att), channel number (Att)	IGNORED		
	appendix b (E), datasheet 1 (E)	TRUE		
	datasheet 1 (E), repositioning (Att)	TRUE		
all personnel working on this procedure have signed the signature record sheet attached in Section 2.5 of this document	personnel (E), signed (Ac)	TRUE	0	3/3
	signature record sheet (E), signed (Ac)	TRUE		
	section 2.5 (E), signature record sheet (E)	TRUE		
FLM independently verify the FME requirements are established	FLM (E), verify (Ac)	TRUE	0	4/4
	FME (E), requirements (E)	TRUE		
	requirements (E), verify (Ac)	TRUE		
	established (V), requirements (E)	TRUE		
tool tethering is mandatory. Also ensure that the catch tray is in place for all repositioning activities	Tool (E), tethering (Ac)	TRUE	1	2/3
	mandatory (V), tethering (Ac)	IGNORED		
	catch tray (E), ensure (Ac)	TRUE		
	repositioning (Att), ensure (Ac)	FALSE		
execution cart is setup and available on both platforms to view specific items as required via custom mounted cameras	execution cart (E), setup (Ac)	TRUE	1	3/4
	setup (Ac), view (Ac)	IGNORED		
	platforms (E), setup (Ac)	FALSE		
	items (E), view (Ac)	TRUE		
	custom mounted cameras (E), items (E)	TRUE		

Figure 5.1: Relations extracted from different types of sentences or iTexts - 1

iText	duplets	relation	wrong	correct/total
ensure equipment, listed in section test equipment tools and consumables, are prepared and ready for use	ensure (Ac), equipment (E)	TRUE	0	4/4
	equipment (E), prepared (Ac)	TRUE		
	test equipment (E), consumables (E)	TRUE		
	consumables (E), equipment (E)	TRUE		
	ready (V), prepared (Ac)	IGNORED		
record tooling calibration data in appendix c	record (Ac), tooling calibration (E)	TRUE	0	2/2
	appendix c (E), tooling calibration (E)	TRUE		
ensure Communicationlinks are setup and tested between Vault and IRI trailer	communication links (E), setup (Ac)	TRUE	0	3/3
	setup (Ac), tested (Ac)	IGNORED		
	tested (Ac), ensure (Ac)	IGNORED		
	vault (E), iri trailer (E)	TRUE		
	iri trailer (E), tested (Ac)	TRUE		
perform function test on both measuring tools, check for binding and smooth operation	perform (Ac), check (Ac)	IGNORED	0	5/5
	function test (E), perform (Ac)	TRUE		
	measuring tools (E), perform (Ac)	TRUE		
	binding (V), operation (E)	TRUE		
	smooth (V), operation (E)	TRUE		
	operation (E), check (Ac)	TRUE		
ensure REP reviewed and initialled	REP (E), reviewed (Ac)	TRUE	0	1/1
	reviewed (Ac), ensure (Ac)	IGNORED		
	initialled (Ac), reviewed (Ac)	IGNORED		
FLM verify all steps completed to this point	FLM (E), verify (Ac)	TRUE	0	3/3
	steps (E), verify (Ac)	TRUE		
	completed (Ac), steps €	TRUE		
pump must have pressure 4 Pa	pump (E), pressure (Att)	TRUE	0	3/3
	pressure (Att), pump (E)	TRUE		
	4 (V), pressure (Att)	TRUE		

Figure 5.2: Relations extracted from different types of sentences or iTexts - 2

iText	duplets	relation	wrong	correct/total
return the crane to the parked position when not in use	return (Ac), crane (E)	TRUE	0	3/3
	crane (E), parked (V)	TRUE		
	position(Att), parked (V)	TRUE		
FME field conditions have been reviewed and discussed with FLM	FME (E), field conditions (E)	TRUE	0	4/4
	field conditions (E), reviewed (Ac)	TRUE		
	discussed (Ac), FLM (E)	TRUE		
	FLM (E), reviewed (Ac)	TRUE		
ensure positioning assembly hardware, quick locknut, p/a stud threads (tube sheet end) and saddle clamp threads have been lubricated with crc penetrating oil	ensure (Ac), lubricated (Ac)	IGNORED	2	3/5
	positioning assembly hardware (E), ensure (Ac)	TRUE		
	quick locknut (E), positioning assembly hardware (E)	FALSE		
	quick locknut (E), positioning assembly hardware (E)	FALSE		
	saddle clamp threads (E), lubricated (Ac)	TRUE		
	crc penetrating oil (E), lubricated (Ac)	TRUE		
ensure personnel, required to access platform, has appropriate current fall arrest qualification	personnel (E), ensure (Ac)	TRUE	0	5/5
	ensure (Ac), personnel (E)	TRUE		
	access (Ac), platform (E)	TRUE		
	platform (E), personnel (E)	TRUE		
	fall arrest (E), ensure (Ac)	TRUE		
ensure stud measurement tool has a valid calibration date	stud measurement (E), ensure (Ac)	TRUE	0	3/3
	valid (V), calibration date (Att)	TRUE		
	calibration date (Att), stud measurement (E)	TRUE		
remove insulation ring on the target channel	insulation ring (E), remove (Ac)	TRUE	0	2/2
	target channel (E), insulation ring €	TRUE		

Figure 5.3: Relations extracted from different types of sentences or iTexts - 3

iText	duplets	relation	wrong	correct/total
shift engineer verify record of target site and confirm tool is secured in position	shift engineer (E), verify (Ac)	TRUE	2	5/7
	verify (Ac), site (E)	TRUE		
	record (Ac), site (E)	FALSE		
	site (E), confirm (Ac)	FALSE		
	tool (E), confirm (Ac)	TRUE		
	secured (V), tool (E)	TRUE		
	position (Att), secured (V)	TRUE		
on the east target site install reconfiguration panlut and unlock target site	site (E), install (Ac)	TRUE	0	4/4
	east (E), site (E)	TRUE		
	reconfiguration panlut (E), install (Ac)	TRUE		
	unlock (Ac), install (Ac)	IGNORED		
	site (E), unlock (Ac)	TRUE		
ensure the torquewrench is set to 180 ft lb	torquewrench (E), set (Att)	TRUE	0	3/3
	set (Att), ensure (Ac)	TRUE		
	180 (V), set (Att)	TRUE		
examine digital indicator on the e face positioning tool to determine any movement of the target channel	examine (Ac), determine (Ac)	IGNORED	1	3/4
	digital indicator (E), examine (Ac)	TRUE		
	e face positioning tool (E), examine (Ac)	FALSE		
	movement (Ac), channel (E)	TRUE		
	channel (E), determine (Ac)	TRUE		
checking the west digital indicator, shift the channel to the east until the required distance has been reached	checking, reached	IGNORED	1	4/5
	west digital indicator (E), checking (Ac)	TRUE		
	shift (Ac), channel (E)	TRUE		
	channel (E), east (E)	TRUE		
	east (E), distance (E)	FALSE		
	distance (E), reached (V)	TRUE		

Figure 5.4: Relations extracted from different types of sentences or iTexts - 4

"IGNORED" means it does not follow. "FALSE" means the relation of the duplet is wrong. 'E', 'Ac', 'Att', and 'V', that is observed in "duplets" column in Figure 5.1, 5.2, 5.3 and 5.4, stands for "Entity", "Action", "Attribute", and "Value", respectively. The term wrong here refers to a relationship that has been identified by our algorithm as valid, but in reality, there is no direct relationship between those terms. Combination of each of these duplets for a particular iText forms a network that is the building block of HESN. Each of these networks is tracked with the help of tags. Finally, the information for each entity, action, attribute, and value is updated, which updates HESN and the knowledge base as a whole.

5.2 Validation of Relation Extraction and HESN

The proposed knowledge base consists of HESN and domain knowledge. HESN is the crucial component of the knowledge base that represents the relationship among different terms based on different operations. This knowledge is represented in the form of nodes and edges. Validation of extracted relations from iText is not a straightforward task because there are no standard criteria for validating represented knowledge from input data. In most cases, the quality of represented knowledge is validated with the help of direct human intervention [38]. Hence, in the case of our proposed knowledge base, the relationship extracted from iText is validated since HESN retains the

relationships.

Standard Information Extraction (IE) metrics (e.g. precision, recall and F-measure) [38] were applied to validate the extracted relationships. Few conditions have been considered when validating the relationships extracted from iText. This is because the relationship extraction also depends on the domain knowledge. The relationships are established only among those terms that have been defined in the domain knowledge. Hence, the validation is done based on the extracted relationships only. When calculating precision, recall and F-measure, the following were considered - True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN) were considered. Equation 5.1, 5.2 and 5.3 shows the formula to calculate precision, recall and f-measure (F1) respectively.

$$Precision = TP/(TP + FP) \tag{5.1}$$

$$Recall = TP/(TP + FN) \tag{5.2}$$

$$F1 = 2 * (Precision * Recall)/(Precision + Recall) \tag{5.3}$$

It is essential to understand how, in this research, a relation is considered as TP, FP, TN or FN. Eventually, these values would help to measure precision and recall. And from precision and recall, F-measure is calculated, which validates the

execution cart is setup and available on both platforms to view specific items as required via custom mounted cameras.	execution cart (E), setup (Ac)	True Positive
	setup (Ac), view (Ac)	True Negative
	platforms (E), setup (Ac)	False Positive
	items (E), view (Ac)	True Positive
	custom mounted cameras (E), items (E)	True Positive

Figure 5.5: Understanding the concept of True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN) in terms of extracted relation for validation of the established relationships among different terms in HESN.

performance of the knowledge retained in HESN. An example has been shown in Figure 5.5. The iText "execution cart is setup and available on both platforms to view specific items as required via custom mounted cameras." consists of some terms that are underlined. The underlined terms are the terms that are defined in the domain knowledge. As a result, relationships among them will be established only. The second column of the table, as shown in Figure 5.5 represents the duplets. Considering that there could be other relationships existing in the same iText, that again depends on the domain knowledge definition. As a result, the validation is done based on these extracted duplets only. The first duplet consisting of the terms "execution cart" and "setup" is considered as a valid relation. It also makes sense from the iText. The relation is also between an Entity (E) and an Action (Ac), which follows one out of six types of relations that HESN allows. As a result, this extracted relationship

is considered as true positive (TP). The same goes for the relationship between the terms "items" and "view", and the terms "custom mounted cameras" and "items". There is a relationship between "setup" and "view". But this relationship is not considered according to our methodology as the relationship is between two action (Ac) terms, and HESN does not allow such relation. Hence, this relationship is considered as true negative (TN) because our methodology predicted it not to be a valid relation, and in reality, from the iText, it is observed that there exists no direct relationship between those two terms. However, our methodology considers a valid relationship between the terms "platforms" and "setup". But from the iText, there exists no direct relationship between them. Hence, this relationship is considered as false positive (FP). A false negative relation would be the relation between two terms predicted as not existing, but in reality, there is a relation between the two terms. A false negative relation or duplet will only exist when a relation between two terms does not follow the six types of relationships mentioned. That means if, for example, there is a relation between an action term such as 'move', 'shift' etc., with any value term such as 'high', 'poor' etc. Since HESN does not allow any relationships other than the six types of relationships mentioned previously, a false negative situation is always ignored. Hence, if a relation is predicted as not valid, that means it is not following the six types of relation criteria of HESN, and it will not be considered

No. of iTexts	No. of Relations Extracted	Precision	Recall	F1 Score
25	102	0.928373016	1	0.957771164

Figure 5.6: iTexts validated with the help of precision, recall and F-measure

anyhow.

Figure 5.5 shows a table representing the result of the validation done on 25 iTexts. 102 relations in total were extracted from these 25 iTexts. The average precision, recall and F-measure were found to be 0.93, 1 and 0.96, respectively. The highest value for precision, recall and F-measure is 1. The closer to 1 the value is, the better the result. Hence, F-measure was found to be 0.96 shows a pretty good result for HESN. The iTexts selected for validation consist of simple to complex sentence structures. The goal here is to extract the maximum number of correct relations and avoid wrong or invalid relations. The validation result help to get an idea about the performance of HESN. Although, this research is not only about extracting and establishing relationships among different terms but also about structuring the knowledge based on different operations and instruction. The extraction of duplets also depends on the way domain knowledge is defined. Hence, it was important to demonstrate with the help of a validation metric to prove the point about mainly avoiding the wrong relations and focusing on how accurate the extracted duplets are.

5.3 Query Evaluation

The knowledge base proposed in this thesis is advantageous when learning from operational and procedural documents that consist of iTexts. The information observed in operational documents, consisting of iTexts, needs to be retrieved based on different operations. Status, condition, parameters, involvement of human role, measurement, activity, etc., varies for different operations, although the terms are the same. That means the name of an entity used in an operation, can be found in another operation too, having different values. Hence, when a query is asked based on an operation, HESN can provide information according to that particular operation. This makes HESN unique and efficient for iTexts. From Figure 3.7, two queries could be considered:

1. What should be the pressure of pump for Operation 1?

2. What should be the pressure of pump for Operation 2?

Here, "Operation 1" and "Operation 2" are the title (PT) of two separate operations. From the title of "Operation 1," we get tags PTAG1. Similarly, from the title of "Operation 2," we get tags PTAG2. Let us consider that both operations has only one instruction or procedure (CT) for each. Then, the tags of CT for "Operation 1"

is T1 = all tags from CT (CTTags1) + “Operation 1” title tags (PTAG1). In the same way, the tags of CT of “Operation 2” is T2 = all tags from CT (CTTags2) + “Operation 2” title tags (PTAG2). Now, it is possible to retrieve the network consisting of the relation among “pressure”, “pump”, and “3” based on T1 for the first query and the network consisting of the relation among “pressure”, “pump” and “7” based on T2 for the second query. In this way, both the questions can be answered using HESN. Moreover, domain knowledge consists of information about the classes of each of the terms. This helps identify entities, actions, attributes and attribute values, and complex reasoning through HESN.answers

5.4 Validation of Qualitative and Quantitative features of HESN

HESN consists of qualitative and quantitative features. Qualitative data talks about the quality of anything, such as the condition of equipment, the colour of any object, etc. [60]. For example, the qualitative value of pressure in the pump could be high or low. It does not have a number in its description. On the other hand, quantitative data provides information with the help of numbers. For example, the quantitative value of pressure in the pump could be 7 pascals.

Each duplet in HESN provides meaningful incite about the two terms and their relation. Let's closely look at figure 3.7. It shows an example of quantitative information retention by HESN as the information clearly states that the pressure of the pump is 3 units in the case of operation 1. HESN also retains information such as the condition of an equipment that could be poor, good, excellent, etc. These are examples of qualitative information.

Chapter 6

Implementation on Industrial Applications

A part of the concept of this research work has been utilized to develop an industrial application called Intelligent Experience Retention System (IERS). It was designed to overcome challenges and limitations of capturing human experience related to operating procedures for plant operation and maintenance in nuclear power plants. It is time-consuming to find specific information from thousands of input documents. Less experienced employees cannot operate complex tasks due to having less knowledge and training about the documents and their operation. Knowledge structure was developed based on HESN to represent inputs from documents, data, text, and even voice related to operation and maintenance instructions in nuclear power plants.

Human experience was captured and integrated within the structured knowledge in an integrated scheme that includes deterministic, qualitative, and probabilistic parameters and attributes that are captured and dynamically tuned throughout the execution of the system. An article [21] has been published that talks about IERS.

6.1 IERS Background

In nuclear power plants, and due to continuous moves of experienced personnel to different department or retirements, a vast amount of expertise and systems-specific knowledge is lost. This leads to longer training periods for new employees and sometimes to delayed responses to problems. These delayed responses can be due to either a lack of knowledge or as a result of the cognitive impact from stress resulting in a physiological response to the stress that prevents clearer decision making. Due to the safety-criticality characteristics of applications, it is quite expected to have a lack of knowledge for new employees on performing activities. The loss of expertise costs nuclear power plants a large amount of money as they have to invest in training less experienced staff, and leads to indirect losses in delayed or wrong activities, in particular in reactor maintenance and inspection activities. Moreover, the unavailability of expert systems limited the ability of staff to perform their tasks effectively. Another challenge and need to retain engineering knowledge is specific

activities such as refurbishment or decommissioning, which are repeated over a long period of time (20-50 years) which did not enable building expertise in these areas, and transferring expertise become important. The challenges include: Retirement of expertise in NPP; Transformation and diversity of technologies in plant operation and maintenance; Technological advancement in communications between human-human, machine-human, brain-machine, human-in-the-loop and machine-machine; and Expensive and time consuming training of young generation plant engineers and personal. Moreover, the unavailability of expert systems limited the ability of staff to perform their tasks effectively. Retrieving information related to any procedure or operation from an uncountable number of documents consisting of countless iTexts is time consuming. Especially during an operation, searching and locating desired instructions from iTexts has to be done quickly and with accuracy without which the entire operation may lead to failure. In addition, there is no proposed effective way to integrate network structure with the knowledge associated with iTexts, with limitation to learn about different operations.

IERS will enable nuclear power plants to manage the challenge of an ageing work-force approaching retirement. IERS will ensure retention of human experience in view of dynamic update of large amount of operating procedures. IERS can also support operation that takes long time such as decommissioning and refurbishment activities,

which will reduce the limitation of retaining human experience over such long period of time. IERS will support nuclear power plants to manage operation and maintenance expertise before the retirement of current expert employees. IERS can also be used to examine the training effectiveness by evaluating the trainee's answers to a number of questions and validate their answers to assess their learning and expertise level as well as evaluate the speed of their response in different situations and scenarios, and in different conditions such as emergencies and stresses. IERS can be used in several applications within nuclear power plant, to enhance a number of functional areas, including maintenance, operation, inspection, and decommissioning activities. The proposed system will enable nuclear power plants to have overall cost savings in terms of operation and maintenance critical time, training time, and decision making effort. It also reduced the human error risks and the dose uptake during operation and maintenance activities. In addition, IERS will increase experience and knowledge of the new employees, and enable knowledge transfer from experienced employees to inexperienced ones, which will lead to reduced operation and maintenance inspection risks, dose, time, and cost. IERS will support the business continuity by transferring knowledge and experience from senior employees to inexperienced ones, and protect business knowledge from being lost. Also it will be used to evaluate different experiences and validate their contents. IERS has been developed and demonstrated using

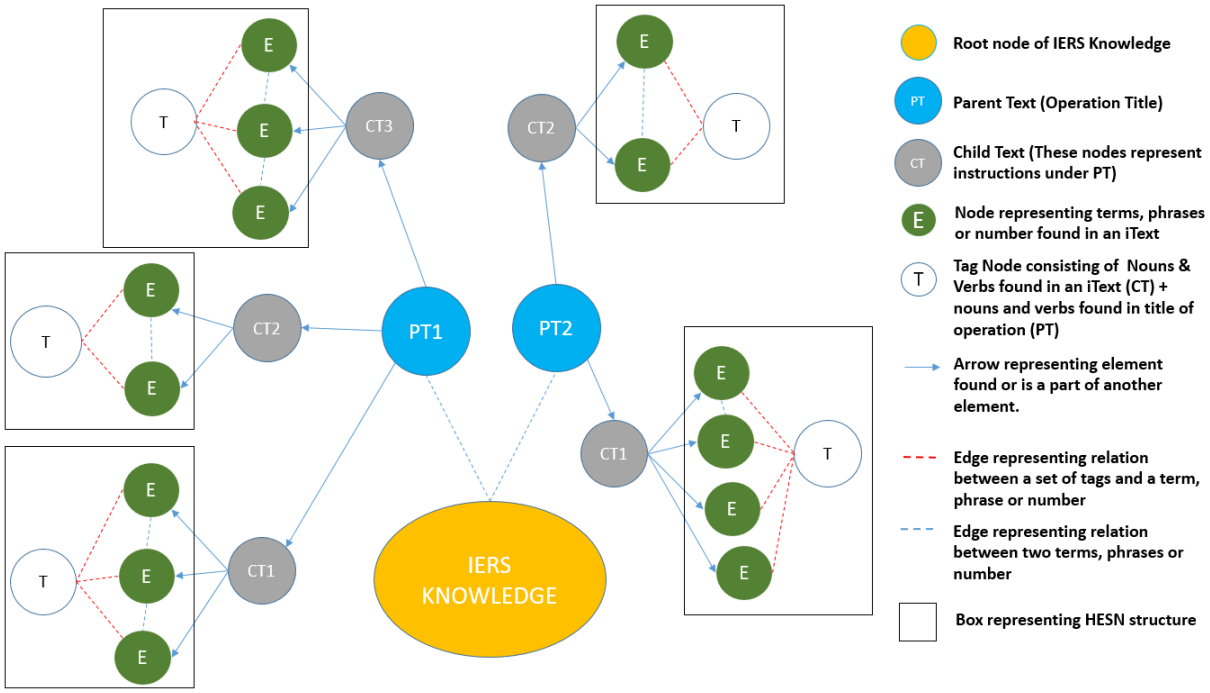


Figure 6.1: Knowledge structure of IERS

number of case studies from nuclear power plants where human experience is captured and associated with the knowledge extracted from input procedures and documents, and used to answer questions, queries, and update the stored knowledge base with learning capabilities.

6.2 Implementation of HESN into IERS

IERS was implemented based on inputs and outputs where multiple input documents were uploaded into IERS and their contents transformed into knowledge and struc-

tured into the knowledge base. The input can also be voice or small text used to help IERS learn about different information or instruction. To capture human experience, a knowledge base was developed. HESN was also a part of the knowledge base. The structure of the knowledge base consisting of HESN was a little bit different in the case of IERS. The knowledge base retains the relationship among different instructions under an operation, whereas HESN retains the relationship among different terms or phrases. IERS is used to train newly hired or inexperienced engineers as well as experts from different domains. IERS learn from input documents and procedures, as well as input data and text or voice, where multiple documents or inputs will be merged into the knowledge base and human experience will be accumulated and integrated within HESN. IERS is an excellent example of a complex man-machine system with human-in-the-loop capabilities, which will address increased human reliability and overall system availability. Moreover, IERS is used to support fault detection, isolation, and diagnosis with the help of the knowledge base and accumulated knowledge during the run of the system. HESN includes nodes that represent the knowledge structure of documents and experience, which include facts, data, rules, and constraints. IERS has the potential to learn from real-time plant data from operation and maintenance, which supports real-time plant and human monitoring. The knowledge structure of IERS knowledge base consisting of HESN is shown in Figure

6.1

6.3 User Interface of IERS

IERS consists of four main user interfaces or pages. User interact using these pages to which helps IERS to capture knowledge, retrieve information, teach new words and delete documents and information. Figure 6.2 is the user interface using which the user upload document and IERS learn procedures from the uploaded document. If there are any unknown terms found in the document, which are not found in the domain knowledge of IERS, then these would also appear here and the user gets to classify and describe those new terms. Figure 6.3 shows how list of unknown terms are defined and classified which were found from the document and are unknown to the system. The system stores the information after confirmation from the user. User ask questions to the system and desired information is retrieved using the user interface as shown in Figure 6.4. User can either ask using voice or input field. In case of using voice, the system wakes up and starts listening when called "Hey IERS". Then the user can ask any question, and the system replies with the desired information. It also provides the reference of the answer. That can be viewed by clicking on the "Reference" button as shown in figure 6.5. It shows the document title and page number. Moreover, it provides other related answers to the desired question too,

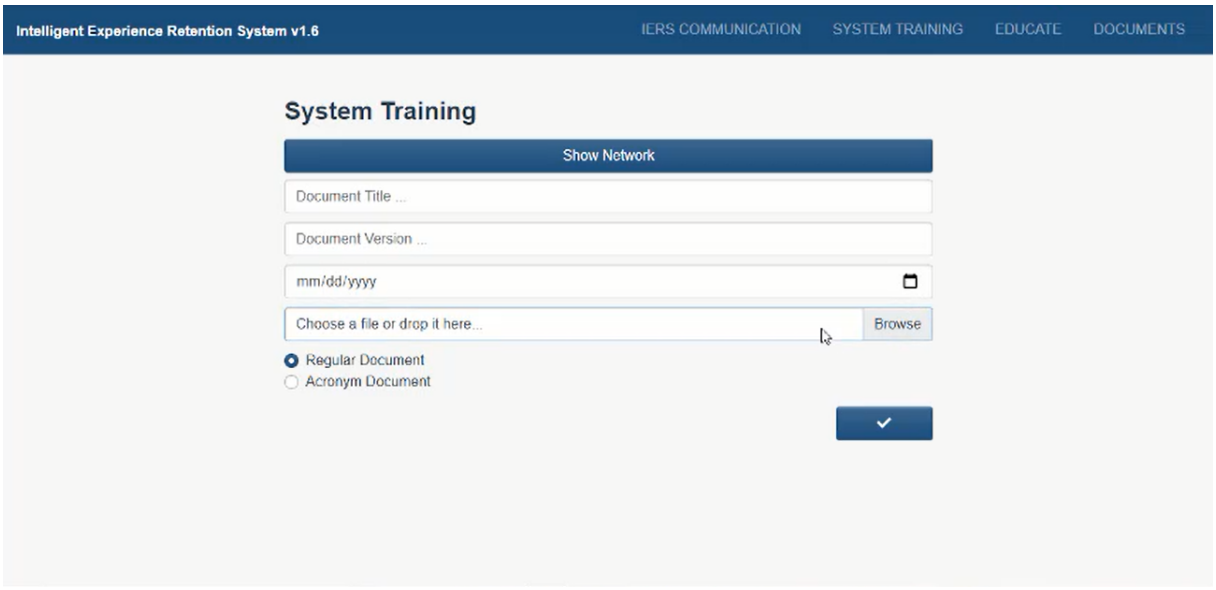


Figure 6.2: IERS - Capture knowledge from documents using this user interface

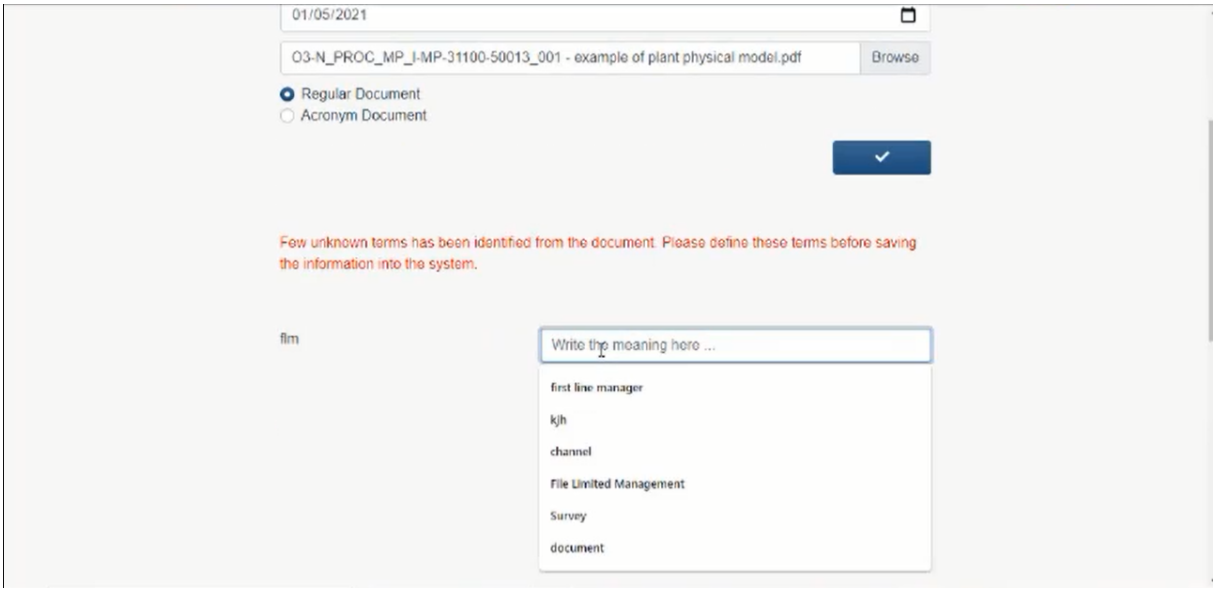


Figure 6.3: IERS - Defining unknown terms found in the document

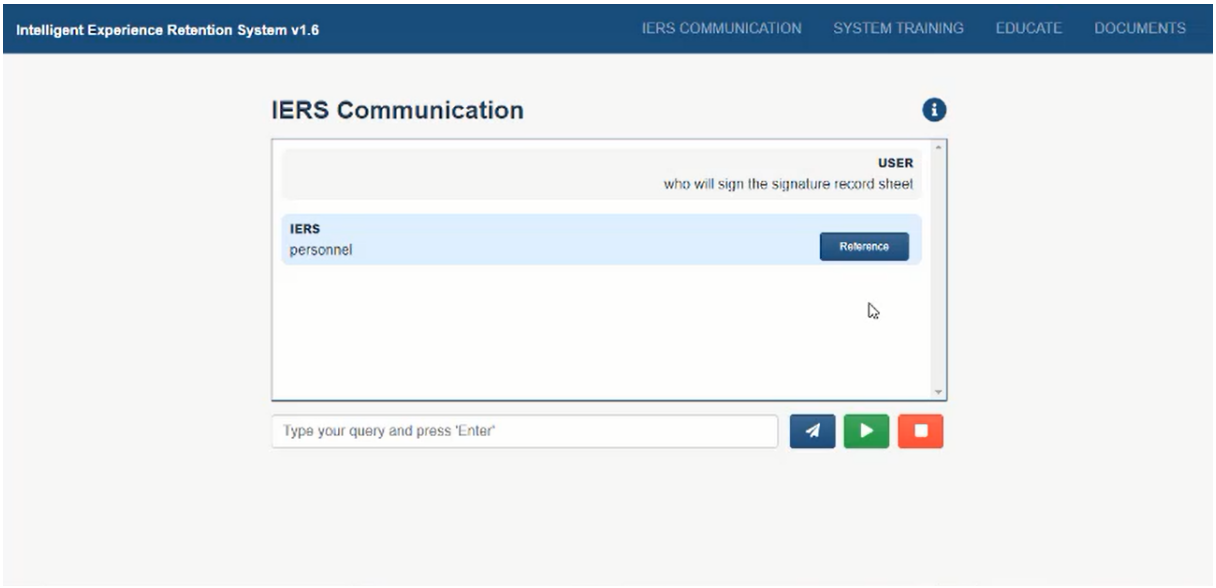


Figure 6.4: IERS - Information Retrieval / System Chatbox User Interface

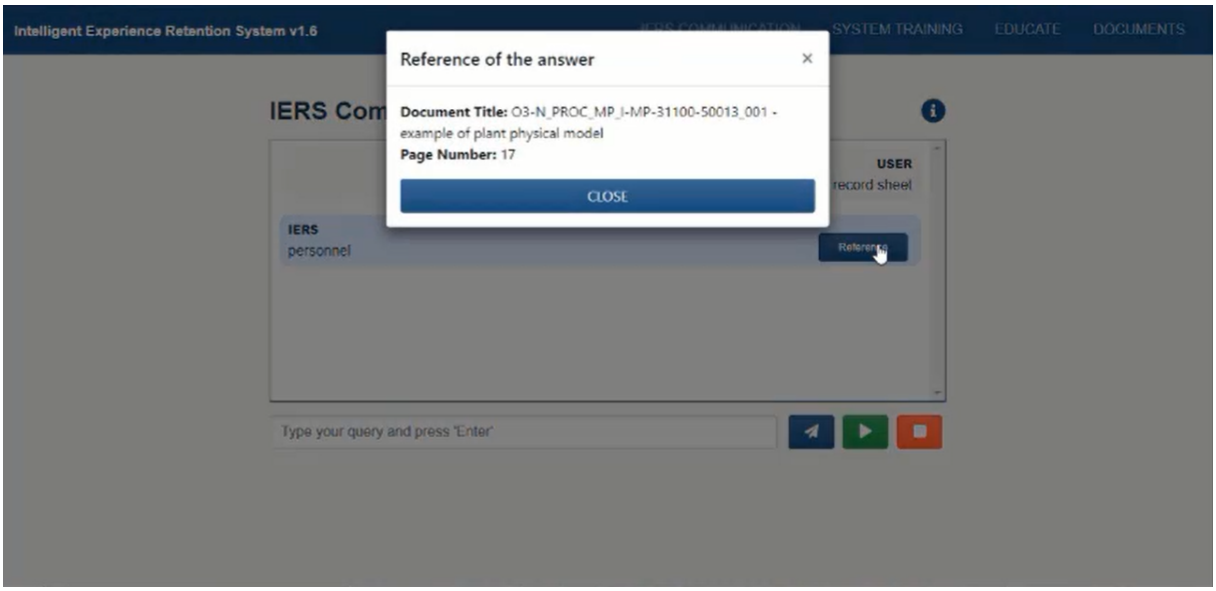


Figure 6.5: IERS - Showing reference of the retrieved answer to know from which document the information is located

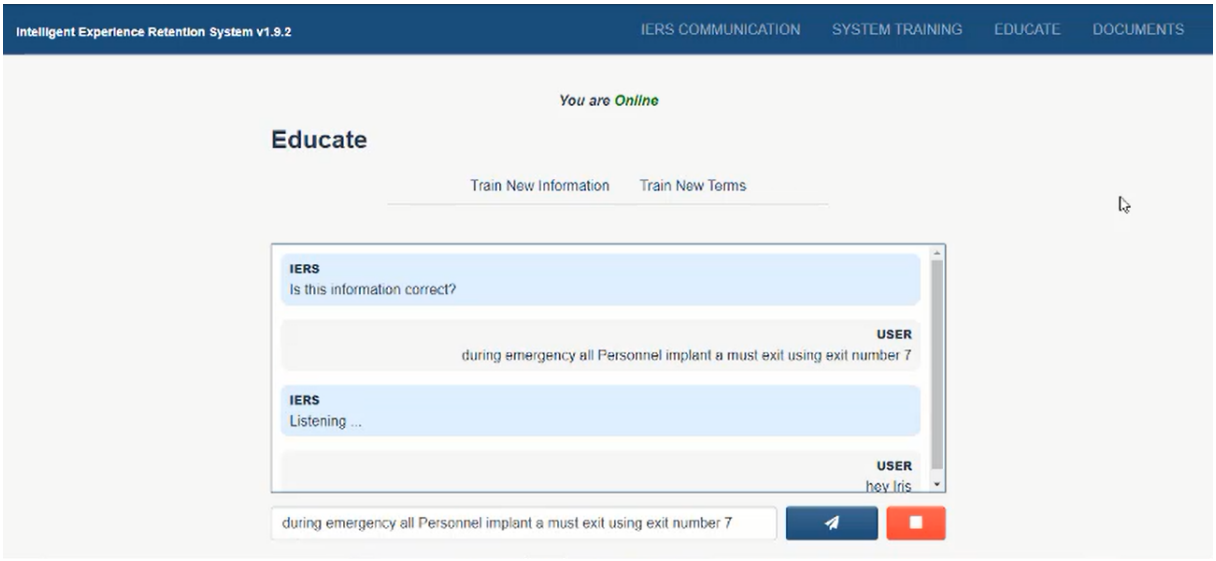


Figure 6.6: IERS - Procedural knowledge learning from direct user communication

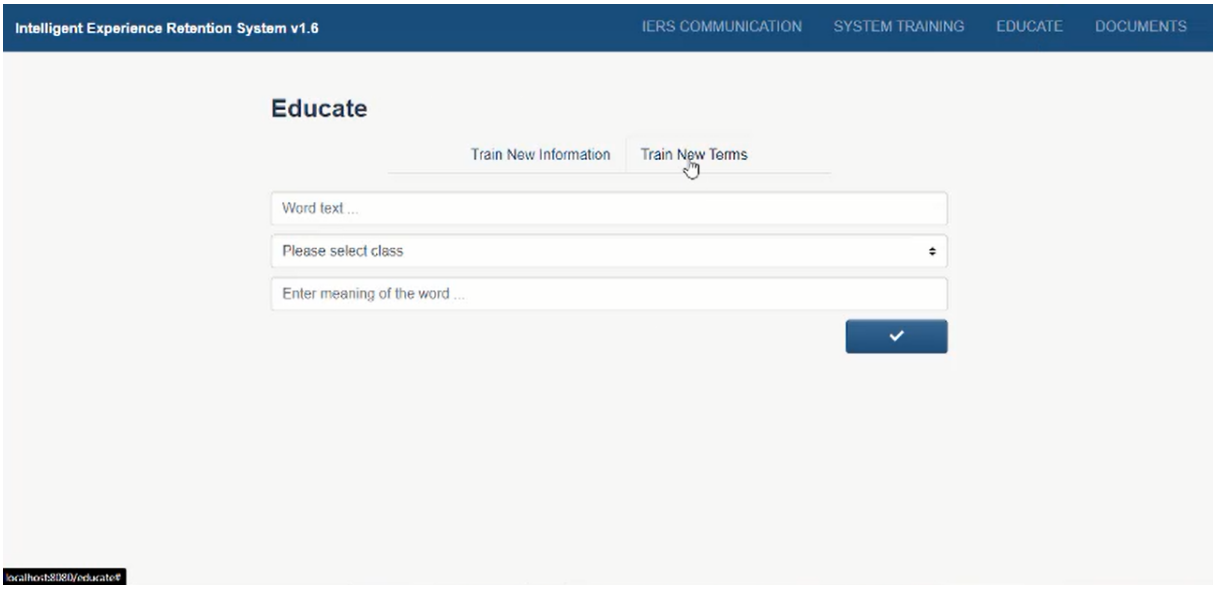


Figure 6.7: IERS - Words and phrases learning from direct user communication

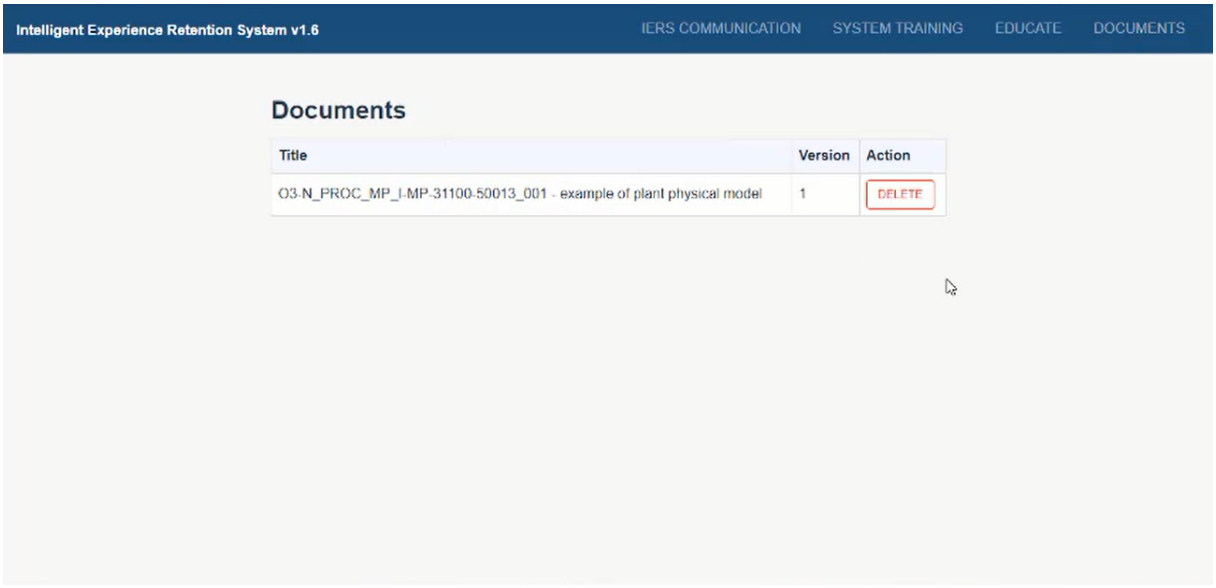


Figure 6.8: IERS - Documents User Interface

when found. But it always replies with the most matched information. There are two more ways of learning the procedures and operations. Firstly, a user can add knowledge to the system with an additional instruction that is not described in any document. This addition of knowledge can also be done using voice. Similarly, the system wakes up when called by "Hey IERS." Afterwards, the desired instruction is spoken, and the system listens to it. The system asks for confirmation. If confirmed, the system stores the information. Otherwise, the user gets to repeat the instruction again. Figure 6.6 shows the user interface of that feature. Secondly, the system can also be taught new words, terms or acronyms by the user. All these knowledge inputs are done using the user interface as shown in Figure 6.7. All these learning

processes of IERS can be done using both voice and text for both cases. Lastly, the document manipulation is done using the documents user interface as shown in Figure 6.8. It shows how many documents were uploaded and what they are. Users can also delete a document from this user interface. By deleting the document, all its related knowledge is deleted from the system.

6.4 Technology used in IERS

IERS implementation includes 3 servers. The User Interface built with Vue.js framework, which is a front end framework developed using JavaScript. It is run using Node.js. Node.js is a JavaScript run time environment. Python Server is used to perform reading and learning from document. A popular library named Flask is used in Python to develop Application Programming Interface (API). Another back-end server, built with Node.js, is used to communicate between User Interface and Database with the help of API. The database that has been used is MySQL. It is a relational database. For the Speech-To-Text and Text-To-Speech part in the system, Google Web Speech API and Deepspeech is used. Google Web Speech API gives more accuracy as it uses data from internet and provides greater accuracy. So it requires internet connection. On the other hand, Deepspeech works without internet but it has less accuracy compared to Google Web Speech API. Both the options are

provided so that the system can perform in both scenarios - with or without internet.

6.5 Key Functions and Features

There are many features and functionalities of IERS. The key features are as follows:

1. Reading from documents and capture human experience, in a structured manner, associated with the industrial operation related to procedures, tools, equipment, documents, location and action mentioned in the documents.

2. Develop a knowledge network, HESN, from the knowledge it acquires from the document and do reasoning to produce accurate output relative to the questions when the IERS is asked.

3. It is capable to filter out the search result based on parameters such as temperature, environment, lab condition etc. This helps to reduce search time and the risk of accident.

4. Receives query using both text and voice. This is helpful for operators who require wearing gloves and at the same time also need to use IERS.

5. IERS learns from documents or small texts. It can also be taught new terms related to the domain knowledge and it updates HESN and improves its reason-

ing. The more it learns, the more it becomes intelligent, the more it can serve well.

6. IERS can automatically classify new terms based on the existing terms in the system.

Chapter 7

Conclusion

Information extraction from iText is not similar to that from regular text or paragraph. It is very important to structure information and relationships of a term or key phrase with other terms on the basis of different operations so that it can be easily identified that what are the set of relations of that term with other terms in case of a particular operation. This research work proposes a knowledge base consisting of Human Experience Semantic Network (HESN) and domain knowledge, which helps to capture the knowledge or human experience from iTexts and dynamically update the knowledge base. The domain knowledge consists of possible terms and key phrases of a particular domain. This helps to identify the target terms in the iText and establish relationships among them in the form of small networks. These small

networks, consisting of relationships among different terms, are also tracked to know from which particular instruction and operation the small network has been formed. All these small networks together form a semantic network which is the HESN itself. The HESN is updated each time new information is learned. The methodology is suitable for extracting knowledge from iText. The current research was focused on iText found in industrial documents from the Nuclear Power Plant domain.

7.1 Research Contribution

The contribution of this research work can be summarized into two main points. These are as follows -

1. The development of a knowledge structure which is adaptive, dynamic and deterministic having qualitative and quantitative attributes. It is called Human Experience Semantic Network (HESN). Knowledge is captured and structured from iTexts with the help of HESN in the form of nodes and edges;

2. The development of a knowledge base having two components. The first one is HESN, which is the main component of the knowledge base. And the second one is the domain knowledge. The knowledge base helps to retain the properties, values, and relationships of different terms or key phrases, found in iTexts.

These terms could sometimes be an entity, or an action term, or an attribute or attribute value. The knowledge that is structured for different entities, action terms, attribute, or attribute values are based on different operations or instructions;

3. The development of a learning algorithm which creates relationship among different terms found in the iText and form a small network, which is the building block of HESN;

7.2 Limitations

Extracting knowledge from text has always been challenging, and there are many popular ways to do that. However, all these techniques lack in extracting knowledge from iText. In this research work, the proposed approach captures knowledge from iText and establishes a knowledge base consisting of HESN and domain knowledge. There are few limitations to the proposed method. In some cases, when an iText consists of too many entities, actions, attributes or attribute values, the duplet formation becomes complex based on the term dependency. Especially if there are entities organized in the sentence one after the other. For e.g. "Record channel number, date, start time, end time, lab number and repositioning required in appendix B data sheet

1". Moreover, the current approach also cannot determine "It," "This," etc., when used in the immediate next sentence. For e.g. "Pump pressure must be checked timely. Without checking it' there could be a leak which may cause a problem." Here, "it" used in the second sentence refers to the "pump pressure." However, this detection is not covered in our approach.

Apart from sentence structure level, a few limitations include assigning terms based on class and category in domain knowledge. For example, a term called 'increase' could be found both as a noun or verb. The noun form of the term 'increase' could be this: 'The water level must have an increase of 2 percent after applying pressure.' The verb form for the same term could be this: 'Increase the pressure till it reaches a satisfactory level'. It is difficult for a domain expert to categorize and classify this term. Usually, this can be categorized as both 'action' and 'value' based on the domain expert. Although the solution has not been provided in this thesis, the challenge could be solved using the same Parts-Of-Speech tagging technique. As the technique identifies how the term is used in the sentence, whether as a noun or verb, its characteristics could be identified based on this information. If the term 'increase' is used as a noun, it is not an action and will be recognized as a value category. On the other hand, if it is a verb, that means it is some sort of action, and the action category would get recognized. Although it may be easy to identify the term whether

used as action or value, it will be challenging in the case of a term categorized as an entity, attribute and value. Since all of these are mostly nouns, classifying and categorizing a term that falls into two or all three of these categories would be tricky. Although this is totally based on the domain expert to classify and categorize in the most acceptable way possible, this could be a task that could be dealt with in future.

7.3 Future Work

Future work includes working with more data and more extensive domain knowledge. Establishing an improved structure of HESN for better representation of relations. Linking relation in a sentence consisting of too many entities, action, attribute and attribute value. Work on improving the technique of classifying and categorizing terms in the domain knowledge for a term that belongs to more than one class of different categories. This classification method must be done in a way so that when iTexts are read, it is possible to identify the correct category or class of the term in the way it is expected as part of iText. Finally, an information retrieval mechanism from HESN based on natural language query.

7.4 Publications

The research presented in this thesis has been published in the journal named "Applied Sciences" under the section "Computing and Artificial Intelligence," belonging to the special issue "Integrating Knowledge Representation and Reasoning in Machine Learning." The published paper [20] equally discusses the knowledge base, HESN and the knowledge capturing methodology from iText.

Part of this research work has also been accepted for publication in IEEE SMC Magazine [21]. It discusses mostly about the software system named Intelligent Experience Retention System (IERS). The discussion is about how HESN and knowledge base are used to develop IERS that reads iText from instructive text and becomes ready to answer related questions. Not only the learning part, but also the information retrieval and query processing part is included in this paper. The system has been discussed in detail in this paper.

References

[1] Spacy.io, industrial-strength natural language processing in python https://spacy.io/ (accessed Apr 18, 2021).

[2] Fumiya Akasaka, Yutaro Nemoto, Koji Kimita, and Yoshiki Shimomura. Development of a knowledge-based design support system for product-service systems. *Computers in Industry*, 63(4):309–318, May 2012.

[3] Mohammed Alliheedi. 2019, Procedurally rhetorical verb-centric frame semantics as a knowledge representation for argumentation analysis of biochemistry articles, PhD thesis, University of Waterloo, Waterloo.

[4] Gonzalo A. Aranda-Corral, Joaquín Borrego-Díaz, Juan Galán-Páez, and Daniel Rodríguez-Chavarría. Towards a notion of basis for knowledge-based systems—applications. *Mathematics*, 9(3):252, January 2021.

[5] Jamshaid Ashraf, Omar Khadeer Hussain, and Farookh Khadeer Hussain. A framework for measuring ontology usage on the web. *The Computer Journal*, 56(9):1083–1101, November 2012.

[6] Nguyen Bach and Sameer Badaskar. A review of relation extraction.

[7] Giannis Bekoulis, Johannes Deleu, Thomas Demeester, and Chris Develder. Joint entity recognition and relation extraction as a multi-head selection problem. *Expert Systems with Applications*, 114:34–45, December 2018.

[8] Pierre Bourhis, Juan L. Reutter, and Domagoj Vrgoč. JSON: Data model and query languages. *Information Systems*, 89:101478, March 2020.

[9] Tianyou Chai, Jinliang Ding, and Fenghua Wu. Hybrid intelligent control for optimal operation of shaft furnace roasting process. *Control Engineering Practice*, 19(3):264–275, March 2011.

[10] Daniel Lage Chang, Jamile Sabatini-Marques, Eduardo Moreira da Costa, Paulo Mauricio Selig, and Tan Yigitcanlar. Knowledge-based, smart and sustainable cities: a provocation for a conceptual framework. *Journal of Open Innovation: Technology, Market, and Complexity*, 4(1), February 2018.

[11] L.P. Coladangelo. 2020, Ontology and domain knowledge base construction for contra dance as an intangible cultural heritage: A case study in knowledge organization of american folk dance, Master's thesis, Kent State University, Kent.

[12] Heather J. Cole-Lewis, Arlene M. Smaldone, Patricia R. Davidson, Rita Kukafka, Jonathan N. Tobin, Andrea Cassells, Elizabeth D. Mynatt, George Hripcsak, and Lena Mamykina. Participatory approach to the development of a knowledge base for problem-solving in diabetes self-management. *International Journal of Medical Informatics*, 85(1):96–103, January 2016.

[13] Gerard Deepak, Naresh Kumar D, and A Santhanavijayan. A semantic approach for entity linking by diverse knowledge integration incorporating role-based chunking. *Procedia Computer Science*, 167:737–746, 2020.

[14] Danilo J. Rezende Diederik P. Kingma, Shakir Mohamed and Max Welling. Semi-supervised learning with deep generative models. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems*, pages 3581–3589, December 2014.

[15] Jinliang Ding, Tianyou Chai, Hong Wang, and Xinkai Chen. Knowledge-based global operation of mineral processing under uncertainty. *IEEE Transactions on Industrial Informatics*, 8(4):849–859, November 2012.

[16] Haihong E, Siqi Xiao, and Meina Song. A text-generated method to joint extraction of entities and relations. *Applied Sciences*, 9(18):3795, September 2019.

[17] Claudio Favi, Roberto Garziera, and Federico Campi. A rule-based system to promote design for manufacturing and assembly in the development of welded structure: Method and tool proposition. *Applied Sciences*, 11(5):2326, March 2021.

[18] Internet Engineering Task Force. The javascript object notation (json) data interchange format, March 2014.

[19] Emmanuel Francalanza, Jonathan Borg, and Carmen Constantinescu. Development and evaluation of a knowledge-based decision-making approach for designing changeable manufacturing systems. *CIRP Journal of Manufacturing Science and Technology*, 16:81–101, January 2017.

[20] Hossam A. Gabbar, Sk Sami Al Jabar, Hassan A. Hassan, and Jing Ren. Development of knowledge base using human experience semantic network for instructive texts. *Applied Sciences*, 11(17):8072, August 2021.

[21] Hossam A. Gabbar, Sk Sami Al Jabar, Hassan A. Hassan, and Jing Ren. An intelligent experience retention system: Challenges and limitations for operation and maintenance in nuclear power plants. 7(4):31–34, October 2021.

[22] ZhiQiang Geng, GuoFei Chen, YongMing Han, Gang Lu, and Fang Li. Semantic relation extraction using sequential and tree-structured LSTM with attention. *Information Sciences*, 509:183–192, January 2020.

[23] Aibo Guo, Zhen Tan, and Xiang Zhao. Measuring triplet trustworthiness in knowledge graphs via expanded relation detection. In *Knowledge Science, Engineering and Management*, pages 65–76. Springer International Publishing, 2020.

[24] Xu Han, Zhiyuan Liu, and Maosong Sun. Neural knowledge acquisition via mutual attention between knowledge graph and text, 2018.

[25] Laura Jeffrey. Different types of text. https://shorturl.at/bcppu (accessed Dec 08, 2021). Apr 2007.

[26] Jovita, Linda, Andrei Hartawan, and Derwin Suhartono. Using vector space model in question answering system. *Procedia Computer Science*, 59:305–311, 2015.

[27] Rajbabu K., Harshavardhan Srinivas, and Sudha S. Industrial information extraction through multi-phase classification using ontology for unstructured documents. *Computers in Industry*, 100:137–147, September 2018.

[28] Bernard Kamsu-Foguem, Fabien Rigal, and Félix Mauget. Mining association rules for the quality improvement of the production process. *Expert Systems with Applications*, 40(4):1034–1045, March 2013.

[29] Mohammad Reza Khosravani, Sara Nasiri, and Tamara Reinicke. Intelligent knowledge-based system to improve injection molding process. *Journal of Industrial Information Integration*, page 100275, September 2021.

[30] Taejin Kim, Yeoil Yun, and Namgyu Kim. Deep learning-based knowledge graph generation for COVID-19. *Sustainability*, 13(4):2276, February 2021.

[31] Agnieszka Konys. Knowledge systematization for ontology learning methods. *Procedia Computer Science*, 126:2194–2207, 2018.

[32] Luoqin Li, Jiabing Wang, Jichang Li, Qianli Ma, and Jia Wei. Relation classification via keyword-attentive sentence mechanism and synthetic stimulation loss. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 27(9):1392–1404, September 2019.

[33] Pengfei Li and Kezhi Mao. Knowledge-oriented convolutional neural network for causal relation extraction from natural language texts. *Expert Systems with Applications*, 115:512–523, January 2019.

[34] Tong Li and Zhishuai Chen. An ontology-based learning approach for automatically classifying security requirements. *Journal of Systems and Software*, 165:110566, July 2020.

[35] Fanjie Lin. 2018, Constructing knowledge graph for cybersecurity education, Master's thesis, Arizona State University.

[36] Oier Lopez de Lacalle and Mirella Lapata. Unsupervised relation extraction with general domain knowledge. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 415–425, Seattle, Washington, USA, October 2013. Association for Computational Linguistics.

[37] Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3219–3232, Brussels, Belgium, October-November 2018. Association for Computational Linguistics.

[38] Jose L. Martinez-Rodriguez, Ivan Lopez-Arevalo, and Ana B. Rios-Alvarado. OpenIE-based approach for knowledge graph construction from text. *Expert Systems with Applications*, 113:339–355, December 2018.

[39] Jose L. Martinez-Rodriguez, Ivan Lopez-Arevalo, and Ana B. Rios-Alvarado. OpenIE-based approach for knowledge graph construction from text. *Expert Systems with Applications*, 113:339–355, December 2018.

[40] Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011, Suntec, Singapore, August 2009. Association for Computational Linguistics.

[41] Marko Mladineo, Marina Crnjac Zizic, Amanda Aljinovic, and Nikola Gjeldum. Towards a knowledge-based cognitive system for industrial application: Case of personalized products. *Journal of Industrial Information Integration*, page 100284, September 2021.

[42] Raghu Chaitanya Munjulury. 2017, Knowledge-based integrated aircraft design, Linköping University, Linköping.

[43] David Nadeau and Satoshi Sekine. A survey of named entity recognition and classification. *Lingvisticæ Investigationes. International Journal of Linguistics and Language Resources*, 30(1):3–26, August 2007.

[44] Binling Nie and Shouqian Sun. Knowledge graph embedding via reasoning over entities, relations, and text. *Future Generation Computer Systems*, 91:426–433, February 2019.

[45] Elpiniki I. Papageorgiou and Jose L. Salmeron. A review of fuzzy cognitive maps research during the last decade. *IEEE Transactions on Fuzzy Systems*, 21(1):66–79, February 2013.

[46] Justyna Patalas-Maliszewska and Sławomir Kłos. Knowledge network for the development of software projects (KnowNetSoft). *IFAC-PapersOnLine*, 51(11):776–781, 2018.

[47] Paolo J. Piunno. 2012, Expert knowledge base development for an industrial energy assessment system, Master's thesis, University of Windsor, Windsor.

[48] Jerzy Pokojski, Konrad Oleksiński, and Jarosław Pruszyński. Knowledge based processes in the context of conceptual design. *Journal of Industrial Information Integration*, 15:219–238, September 2019.

[49] Atul Prakash Prajapati and D.K. Chaturvedi. Semantic network based knowledge representation for cognitive decision making in teaching electrical motor concepts. In *2017 International Conference on Computer, Communications and Electronics (Comptelix)*. IEEE, July 2017.

[50] Yongbin Qin, Weizhe Yang, Kai Wang, Ruizhang Huang, Feng Tian, Shaolin Ao, and Yanping Chen. Entity relation extraction based on entity indicators. *Symmetry*, 13(4):539, March 2021.

[51] Petar Ristoski, Anna Lisa Gentile, Alfredo Alba, Daniel Gruhl, and Steven Welch. Large-scale relation extraction from web documents and knowledge graphs with human-in-the-loop. *Journal of Web Semantics*, 60:100546, January 2020.

[52] Mathieu Ritou, Farouk Belkadi, Zakaria Yahouni, Catherine Da Cunha, Florent Laroche, and Benoit Furet. Knowledge-based multi-level aggregation for decision aid in the machining industry. *CIRP Annals*, 68(1):475–478, 2019.

[53] Miguel Ángel Rodríguez-García, Francisco García-Sánchez, and Rafael Valencia-García. Knowledge-based system for crop pests and diseases recognition. *Electronics*, 10(8):905, April 2021.

[54] Emilio M. Sanfilippo, Farouk Belkadi, and Alain Bernard. Ontology-based knowledge representation for additive manufacturing. *Computers in Industry*, 109:182–194, August 2019.

[55] Petr Olegovich Skobelev, Elena V. Simonova, S.V. Smirnov, Denis S. Budaev, George Yu Voshchuk, and A.L. Morokov. Development of a knowledge base in

the "smart farming" system for agricultural enterprise management. *Procedia Computer Science*, 150:154–161, 2019.

[56] Marco Antonio Calijorne Soares and Fernando Silva Parreiras. A literature review on question answering techniques, paradigms and systems. *Journal of King Saud University - Computer and Information Sciences*, August 2018.

[57] John F. Sowa. Semantic networks, 1987.

[58] Lu Tang, Bijie Bie, and Degui Zhi. Tweeting about measles during stages of an outbreak: A semantic network approach to the framing of an emerging infectious disease. *American Journal of Infection Control*, 46(12):1375–1380, December 2018.

[59] Xing Tang, Ling Chen, Jun Cui, and Baogang Wei. Knowledge representation learning with entity descriptions, hierarchical types, and textual relations. *Information Processing & Management*, 56(3):809–822, May 2019.

[60] Courtney Taylor. What is qualitative data?. shorturl.at/cxlnq (accessed Dec 18, 2021). Jan 2019.

[61] Duc-Thuan Vo and Ebrahim Bagheri. Feature-enriched matrix factorization for relation extraction. *Information Processing & Management*, 56(3):424–444, May 2019.

[62] Zheng Wang, Shuo Xu, and Lijun Zhu. Semantic relation extraction aware of n-gram features from unstructured biomedical text. *Journal of Biomedical Informatics*, 86:59–70, October 2018.

[63] Jaroslaw Watrobski. Ontology learning methods from text - an extensive knowledge-based approach. *Procedia Computer Science*, 176:3356–3368, 2020.

[64] Jaroslaw Watrobski and JaroslawJankowski. Knowledge management in MCDA domain. In *Proceedings of the 2015 Federated Conference on Computer Science and Information Systems*. IEEE, October 2015.

[65] Chengke Wu, Peng Wu, Jun Wang, Rui Jiang, Mengcheng Chen, and Xiangyu Wang. Ontological knowledge base for concrete bridge rehabilitation project management. *Automation in Construction*, 121:103428, January 2021.

[66] Chenyan Xiong. 2016, Knowledge based text representations for information retrieval, PhD thesis, Carnegie Mellon University, Pittsburgh.

[67] Bei Xu and Hai Zhuge. The influence of semantic link network on the ability of question-answering system. *Future Generation Computer Systems*, 108:1–14, July 2020.

[68] Bei Xu and Hai Zhuge. The influence of semantic link network on the ability of question-answering system. *Future Generation Computer Systems*, 108:1–14, July 2020.

[69] Adam Zagorecki and Marek J. Druzdzel. Knowledge engineering for bayesian networks: How common are noisy-MAX distributions in practice? *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 43(1):186–195, January 2013.

[70] Bin Zhang, Chunhua Yang, Hongqiu Zhu, Peng Shi, and Weihua Gui. Controllable-domain-based fuzzy rule extraction for copper removal process control. *IEEE Transactions on Fuzzy Systems*, 26(3):1744–1756, June 2018.

[71] Shubin Zhao and Ralph Grishman. Extracting relations with integrated information using kernel methods. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics - ACL '05*. Association for Computational Linguistics, 2005.

[72] Yunfei Zhao and Carol Smidts. A method for systematically developing the knowledge base of reactor operators in nuclear power plants to support cognitive modeling of operator performance. *Reliability Engineering & System Safety*, 186:64–77, June 2019.

[73] Weimin Zhong, Chaoyuan Li, Xin Peng, Feng Wan, Xufeng An, and Zhou Tian. A knowledge base system for operation optimization: Design and implementation practice for the polyethylene process. *Engineering*, 5(6):1041–1048, December 2019.