

**SEMANTIC CONCEPT RECOGNITION FROM
STRUCTURED and UNSTRUCTURED INPUTS WITHIN
CYBER SECURITY DOMAIN**

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF INFORMATICS
OF MIDDLE EAST TECHNICAL UNIVERSITY

BY
ALP GÖKHAN HOŞSUCU

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE
DEGREE OF MASTER OF SCIENCE
IN
THE DEPARTMENT OF INFORMATION
SYSTEM

FEBRUARY
2015

**SEMANTIC CONCEPT RECOGNITION FROM STRUCTURED
and UNSTRUCTURED INPUTS WITHIN CYBER SECURITY
DOMAIN**

Submitted by **Alp Gökhan Hoşsucu** in partial fulfillment of the requirements
for the degree of **Master of Science in the Department of Information
Systems**,

Middle East Technical University by,

Prof. Dr. Nazife Baykal _____

Director, Informatics Institute _____

Prof. Dr. Yasemin Yardımcı Çetin _____

Head of Department, Information Systems _____

Prof. Dr. Nazife Baykal _____

Supervisor, Information Systems, METU _____

Examining Committee Members

Prof. Dr. Nazife Baykal _____

IS, METU _____

Dr. Ali Arifoğlu _____

IS, METU _____

Assist. Prof. Dr. Aysu Betin Can _____

IS, METU _____

Assoc. Prof. Dr. Sevgi Özkan _____

IS, METU _____

Dr. Emrah Tomur _____

İYTE _____

Date: 06.02.2015

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

**Name, Last name: Alp Gökhan
Hoşsucu**

Signature

ABSTRACT

SEMANTIC CONCEPT RECOGNITION FROM STRUCTURED and UNSTRUCTURED INPUTS WITHIN CYBER SECURITY DOMAIN

Hoşsucu, Alp Gökhan

M.S Department of Information
Systems

Supervisor: Prof. Dr. Nazife
Baykal

February2015, 93Pages

Linked data initiative has been quite successful in terms of publishing and interlinking data over ontological structures. The success is due to answering semantically rich queries over highly structured data. The utilization of linked data structures are widely used in various domains to solve the problem of producing domain specific knowledge which can be interpreted by automated agents without any human interference. Cyber security field is one of the domains that suffer from the excessiveness of the raw data and lacking of the knowledge which constantly requires incorporation of subject matter experts in security analyzes or reasoning processes. The principle aim of this study is to propose an automated approach for cyber-security related knowledge base generation from scratch by utilizing from both structured and unstructured domain related data. The proposed approach is based on the automatic extraction of significant phrases and conversion of them into semantic concepts within the scope of already existing cyber security databases CWE, CPE, VVS and CCE. The system utilizes this raw data, differentiates the structured and unstructured parts which are processed in different modules for knowledge extraction. These concepts are represented in RDF format which includes all the relationships between entities to construct ontology for cyber security domain. To enhance the knowledge extraction process, NLP oriented approaches including Key Phrase Extraction methodologies are used and data augmentation techniques are applied to the concepts by interlinking them to the entities in Freebase and Wikipedia indexes. As a consequence of these operation series, a modular system is developed which is capable of extracting knowledge from the given cyber security

related data. This accumulated knowledge constitutes a basis for cyber-security ontology which can be used for further vulnerability identification and prevention.

Keywords: Cyber Security Information Extraction, Ontology Development, Linked Data, Data Augmentation, Semantic Concept Extraction

ÖZ

SİBER GÜVENLİK ALANINDA YAPISAL VE DÜZ METİNDEN ANLAMSAL KONSEPT ÇIKARIMI

Hoşsucu, Alp Gökhan

Yüksek Lisans, Bilişim Sistemleri
Bölümü

Tez Yöneticisi: Prof. Dr. Nazife Baykal

Şubat 2015, 93 Sayfa

Ontolojik yapılar üzerinden veri yayını ve veri ilişkilendirilmesi oldukça başarılı olmuştur. Başarı son derece yapılandırılmış veriler üzerinde anlamsal olarak zengin sorguları cevaplayabilmekten kaynaklanmaktadır. Oldukça yaygın olan bu bağlantılı veri yapılarının kullanımı siber güvenlik alanı içinde önemli yer tutmaktadır. Bu tez çalışmasının ana odak alanı siber güvenlik sınırları içerisinde bağlantılı veri yapısını kullanmak ve girilen metin bilgilerinden bu alana özgü bilgi çıkartıp anlamsallaştırmaktır. Sistemin temel çalışma ilkesi girilen metin içerisinde Ortak Zayıflık Numaralandırma (CWE), Ortak Platform Numaralandırma (CPE), Ortak Konfigürasyon Numaralandırma (CCE), Zayıflık Üretici İfadeleri (VVS) ile birlikte Milli Güvenlik Açığı Veritabanı (NVD) gibi var olan siber güvenlik veritabanlarından yararlanarak önemli ifade bulmak ve anlamsal kavramlara dönüştürmektir. Çıkarılan bu kavramlar siber güvenlik alanı içindeki varlıklar arasındaki tüm ilişkileri içerip RDF veri yapısı şeklinde temsil edilmiştir. İşlem yapılrken doğal dil işleme, anahtar sözcük çıkartma gibi yöntemler uygulanmıştır. Verilerin içeriğini geliştirmek amacıyla, Freebase, DBpedia graf veritabanları ve Wikipedia indeksleri kullanılarak ham veri, bilgiye dönüştürülmüştür. Bu operasyonların sonucu olarak, yapısal ya da yapısal olmayan herhangi bir metin kaynağı, güvenlik bağlamı içerisinde yorumlanır ve önceliklendirilebilecektir. Bu bilgi daha sonra güvenlik açılarını belirlemek ve önlemek için kullanılabilinecektir.

Anahtar Kelimeler: Siber Güvenlik Bilgi Çıkartımı, Ontoloji Geliştirme, Bağlantılı Veriler, Bilgi Zenginleştirme, Anlamsal Konsept Çıkarımı

Dedicated to my parents Hikmet & Belgin Hoşsucu for their infinite support and my source of inspiration Meltem who gave me the courage and determination for finishing this thesis.

ACKNOWLEDGEMENT

I would like to thank my thesis advisor Professor Nazife Baykal who canalized me to this study and kept motivated in each particular phase of the work. Her continuous encouragement to progress and positive approach leads this work toward high standards.

I also want to thank my former colleagues Halil Ayyıldız and Siyamed Sinir who laid the foundations of my knowledge in the fields of this study and Nurullah Gürcan who did not refuse to give help for this study.

TABLE OF CONTENTS

ABSTRACT.....	v
ÖZ.....	vii
ACKNOWLEDGEMENT.....	x
LIST OF FIGURES.....	xiii
LIST OF TABLES.....	xv
LIST OF EQUATIONS	xvi
ABBREVIATIONS	xvii
1. INTRODUCTION	1
1.1 Motivation & Background.....	1
1.2 Problem Definition.....	4
2. LITERATURE REVIEW	7
2.1 CyberSecurity Ontology Modeling	7
2.1.1 Manual Approach.....	8
2.1.2 Automatic or Semi-Automatic Approach.....	9
2.2 Semantic Concept Extraction	13
3. METHODOLOGY	17
3.1 Overview	17
3.2 System Architecture.....	18
3.3 Automatic Data Feed Collector.....	19
3.3.1 Cyber Security Related Data Analysis	20
3.3.1.1 Vulnerability Feeds.....	22
3.3.1.2 Vulnerability Vendor Statements.....	23
3.3.1.3 Common Product Enumeration	24
3.3.1.4 Common Configuration Enumeration	25
3.3.2 Input Collection and Differentiation	29
3.4 Structured Data To Relational Object Mapping.....	32
3.4.1 XML To Object Tree Conversion.....	33
3.5 Automatic Triplet Generator From Relational Objects.....	39
3.5.1 Triple Generation Algorithm from Relational Objects	40
3.6 NLP Oriented Information Extractor	44
3.6.1 Information Extraction From Unstructured Inputs.....	45
3.6.2 Phrase Scoring.....	50

3.6.2.1 Text Based Statistical Measurements	51
3.6.2.1.1 Wikipedia Data Source Utilization.....	55
3.6.2.2 Graph Based Measurements.....	59
3.6.2.2.1 Freebase RDF Graph Construction and Utilization.....	59
4. SYSTEM EVALUATION	65
4.1 Evaluation of Knowledge Base from Structured Data.....	65
4.2 Evaluation of Knowledge Base from Unstructured Data	67
5. CONCLUSION	73
5.1 Contribution & Discussion.....	73
Input Data Collection.....	73
RDF Generation From Relational Objects	74
NLP Oriented Knowledge Extraction	75
5.2 Limitations & Future Research	77
REFERENCES	79
APPENDICES.....	82
APPENDIX - A TRIPLE GENERATOR ALGORITHM HELPER METHODS	83
APPENDIX - B DBpedia ONTOLOGY	87
APPENDIX - C FREEBASE ONTOLOGY.....	89
APPENDIX - D APACHE SolR.....	91

LIST OF FIGURES

Figure 1 - Network of Everything	3
Figure 2 - Layered Representation of Knowledge.....	5
Figure 3 - General Block Diagram of the System	17
Figure 4 - Overall Process Diagram of the System.....	19
Figure 5 - Example CVE Data Feed	22
Figure 6 - Example VVS Data Feed	24
Figure 7 - Example CPE Data Feed	25
Figure 8 - Example CCE Data Feed	27
Figure 9 - Configuration Management Control Family	28
Figure 10 - Configuration Settings for CM-6	28
Figure 11 - CVE XML Data Feeds with XSD Schemas.....	30
Figure 12 - Block Diagram for Input Collector Module	31
Figure 13 - Block Diagram for Object Relational Mapper Module.....	33
Figure 14 - XSD File for NVD Vulnerability Data.....	34
Figure 15 - JAXB Binding Configuration	35
Figure 16 - Generated Relational Objects Java Package Structure.....	35
Figure 17 - Generated Vulnerability Java Classes	36
Figure 18 - Generated NVD Java Object Class.....	37
Figure 19 - Generated Java Object Class for Vulnerability.....	38
Figure 20 - JAXB XML Unmarshalling Code Snippet.....	38
Figure 21 - Automatic Triplet Generator From Relational Objects Block Diagram.....	40
Figure 22 - Class Diagram for Vulnerability Type	41
Figure 23 - RDF Model for Vulnerability Type	41
Figure 24 - RDF Triple Generator Algorithm From Relational Objects	43
Figure 25 - Information Extraction Module Block Diagram	46
Figure 26 - Semantic Parser Algorithm Pseudo Code Snippet	49
Figure 27 - RDF Representation of Parsed Vulnerability Summary	50
Figure 28 - Phrase Scoring Block Diagram	51
Figure 29 - Wikipedia Index Utilization Block Diagram.....	57
Figure 30 - Schema for Wikipedia Indexing in SolR	58
Figure 31 - SolR Admin Panel for Wikipedia Index Results	58
Figure 32 - N-Triples RDF Representation of Freebase Linked Data.....	60
Figure 33 - XML Representation of N-Triples Format	60
Figure 34 - Knowledge Representation of the Concept "Denial of Service Attack" in Freebase Graph	62
Figure 35 - Page Rank Algorithm Illustration	63
Figure 36 - Helper Method for Retrieving All Fields In Object.....	83
Figure 37 - Helper Method for Field Conversion	84
Figure 38 - Helper Methods for RDF Model Population	85
Figure 39 - Helper Method for Printing RDF Statements.....	86

Figure 40 - Ontologies Used by DBpedia	87
Figure 41 - Freebase Domains.....	89

LIST OF TABLES

Table 1 - Cyber Security Information Sources with Organizations	21
Table 2 - Part of Speech Tag Definitions	47
Table 3 - Text Based Statistical Measurements	53
Table 4 - Extracted Relations from Structured CVE 2014 Data Feeds	66
Table 5 - Index Based Parameter Set for System Evaluation	69
Table 6 - Graph Based Parameter Set for System Evaluation.....	70
Table 7 - Primitive Statistical Term Measurement Set for System Evaluation	70
Table 8 - Evaluation Results of Index Scoring Based Parameter Set.....	71
Table 9 - Evaluation Results of Graph Based Parameter Set	71
Table 10 - Evaluation Results of Primitive Statistical Term Measurement Set.....	71

LIST OF EQUATIONS

Equation 1 - Text Frequency Calculation Formula	54
Equation 2 - Term Frequency–Inverse Document Frequency Calculation Formula	54
Equation 3 - Inverse Document Frequency Calculation Formula.....	54
Equation 4 - PageRank Calculation Formula	64
Equation 5 - Precision and Recall Calculations.....	67
Equation 6 - F-Measure Calculation Formula.....	67
Equation 7 - Precision and Coverage Calculations with Correctness and Relevance Perspectives	68

ABBREVIATIONS

GCN	Government IT Outcomes Resource
NCCIC	National Cybersecurity and Communications Integration Center
ICS-CERT	Industrial Control Systems Cyber Emergency Response Team
SCAP	Security Content Automation Protocol
NIST	National Institute of Standards and Technology
SPARQL	SPARQL Protocol and RDF Query Language
ARQ	A SPARQL Processor for Jena
NLP	Natural Language Processing
OWL	Web Ontology Language
CVE	Common Vulnerability Enumeration
CPE	Common Product Enumeration
CCE	Common Configuration Enumeration
VVS	Vulnerability Vendor Statements
CVSS	Common Vulnerability Scoring System
OVAL	Open Vulnerability and Assessment Language
CAPEC	Common Attack Pattern Enumeration and Classification
MAEC	Malware Attribute Enumeration and Characterization
WSD	Word Sense Disambiguation
LOGS	Lightweight universal Ontology Generation and Operating Architectures
ARF	ASEAN Regional Forum
POS	Part of Speech
NVD	National Vulnerability Database
RDF	Resource Description Framework
XML	Extensible Markup Language
ITU-T	ITU Telecommunication Standardization Sector
XCCDF	Extensible Configuration Checklist Description Format
XSD	XML Schema Definition
IT	Information Technology
JSON	JavaScript Object Notation
RSS	Rich Site Summary
ORM	Object Relational Mapping
BNF	Backus–Naur Form
UTF-8	Universal Character Set + Transformation Format—8-bit

CHAPTER 1

INTRODUCTION

This chapter presents the motivation behind this study, background of the ideas and a comprehensive definition of the addressed problem in the following sections.

1.1 Motivation & Background

While hot topics in software technologies and approaches are changing from year to year, top level topics remain strictly bound to the security issues. The global tendency in software field is directed by the scalability and elasticity essentials. The number of software that executes standalone is decreasing. In contrast, integrated systems run on controlled and uncontrolled networks are increasing which are extremely charming for the opportunistic cybercriminals. In parallel to this formation, especially primary industrial targets which are financial organizations, petroleum and transportation industries and governmental institutions allocate serious amounts of resource to mitigate the risks of potential cyber threats [\[24\]](#). As a consequence of this, many security domain experts are trained and software tools are developed as well. Especially organizations that have critical software processes and sensitive data are forming cyber security teams with domain experts who are responsible for the general security of virtual systems against intrusions.

There exist different kinds of intrusion detection and preventions systems. The majority of these systems are based on the detection and warning procedures that are composed of a set of predefined rules. Event or anomaly detection tools processes the logs that are produced by the devices within the network and seeks for a correlation between pre-defined rules and the event content. In case of a match, they notify the specified end points. The intersection of these tools is the requirement of the expert staff for effective usage. Since, a system may have faced with numerous threats which may be resulted by the combination of different vulnerabilities that falls out of the defined patterns.

Another point is the rapid change of software versions, malwares and attack patterns which makes rule based tools useless in particular circumstances. Cyber security systems should continually updated and monitored to maintain integrity [22]. In case of a system breach, experts need to define the threats and find ways to prevent them. To perform this, evolved attack patterns should be analyzed very carefully in conjunction with the software updates. Even if every condition is seemed to have fulfilled, error factor of the human based systems should be considered which may cause catastrophic results for a fault intolerant system. From the perspective of the GCN blog the issue is concluded with the following words; *“Doubtless, more effective use could be made of existing budget and staff, but it is unlikely that personnel for effective 24/7 analyst staffing in government SOCs will be available soon. To fill this gap, there will have to be greater reliance on automation rather than humans for the time being.”* [23]

In the light of the information above, it can be clearly stated that there is a substantial need for more sophisticated security automations which are able to act intelligently to support security experts and ease the burden of institutions struggle against the cybercrime. In order to achieve this, the essential need is the standardization of the cyber security related data. There are varieties of sources which can be found on web about vulnerabilities of systems, malwares, software bugs, version incompatibilities etc... There are ongoing projects performed by NCCIC, ICS-CERT, SCAP, MITRE Corporation and NIST which standardize the security artifacts by collection information from different reliable sources. They have a system that can be externally used by individuals to accumulate the data. The enumeration of this data is very significant for security experts, since they can find the information easily and are able to follow the updates. This is a good starting point for the security automation however it still needs human intervention and is not sufficient for machines to read and perform analyses on the data. At this point, the need for the cyber security related knowledge can be perceived more robustly.

Cyber Security knowledge is the key accelerator for the development of security automation systems. It is the initial source for all systems that can add value to the common state of the art and lay the foundations of intelligent intrusion detection/prevention systems. The automated conversion of existing information to knowledge is one of the major problems to solve in this pathway. Knowledge base representation and reasoning replaces the human power with the machines that constitute the basics of artificial intelligence field. Contemporarily, ontological structures are used as

knowledge bases for the expert systems. Since ontology is a formal structure for defining concepts with their interlinked relations, it has the ability to give the automated systems an opportunity to make inferences for different cases.

The motivation behind the usage of semantic web techniques and ontology for addressing the cyber security automation problem relies on the evolution of the internet. It starts from “*network of computers*” and directs toward the concepts of “*network of documents*”. The evolution will continue to “*network of everything*”.[25] The current step through network of everything is “Semantic Web” and “Linked Data” where we can think them as “*network of data*”. The phenomenon is illustrated in the following figure 1;

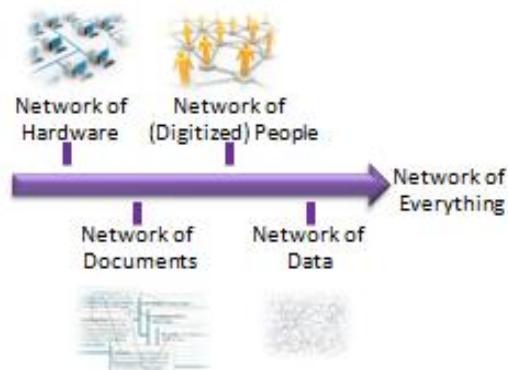


Figure 1 - Network of Everything

The dream of the Semantic Web is: Unlike the current Internet which is based solely on interlinked document, create a network of data and information, where the data and information is machine readable. As the current systems are improved by embracing the semantic approach, automation degree of processes will increase and human dependability will no longer constitutes a problem in fault tolerant systems.

As it is mentioned in the earlier paragraphs, the motivation for this study is to contribute the full automation of cyber security systems. Proposing an end to end solution exceeds the scope of this study, however the essential need for the cyber security related knowledge is addressed in this work with an automatic approach. Since manual ontology modeling methodologies or enhancement of existing cyber security knowledge bases does not meet with the requirements of this general problem, the aim the is minimizing the manual procedures in knowledge generation processes.

1.2 Problem Definition

Transformation of information to knowledge is a problem even in manual approaches. There is a strong need for robust reference knowledge for inference making processes. The automated version of this approach leads to different tasks to resolve and puts the problem one step further. There exists multiple stages of the problem which should be analyzed from different perspectives; otherwise the whole effort may be in vain. The information referred in this problem should be belong to the related domain. Therefore selection of the data is a critical phase while generating a knowledge base. Irrelevant data should be cut off as much as possible at the very initial step of the whole process. The further elimination can be performed by making relative calculations in terms of relevancy. The desired information sources may include data that is structured, unstructured or both. Differentiation of this data is needed since; taking only the structured part will be never sufficient for the knowledge extraction phase. Especially the unstructured cyber security data includes relations, tricks and the necessary input for a knowledge representation. Since, there exists many domain experts which are interpreted the events differently and transfers their experience without adjusting any existing model. Structured and unstructured data should be well analyzed and represented in a compound manner.

The issue related to knowledge base modeling constitutes the next stages of the problem. Defining a static model for the knowledge base which is the ontology in this case does not solves the problem in a sustainable manner, since the static model would only work with predefined data sources and no one can guarantee that these source schemas will not be changed in time. Therefore, in any minor data change, the knowledge extraction phase does not continue to do its task since the system is statically mapped to the defined knowledge base model. Therefore a dynamic approach should be considered within the scope of this problem which automatically maps the given any source of data to the knowledge base without any requirement of manual modification. The scope of the problem perfectly fits to the “Ontology Learning” topic which is addressed in the book [\[25\]](#) as follows; “Ontology learning that is inherently multidisciplinary due to its strong connection with the Semantic Web”. These disciplines are including knowledge representation, logic, philosophy, databases, machine learning, natural language processing, image processing, etc. The components of the ontology learning task are represented in the following figure [\[25\]](#) flows from the basic to the complex formations.

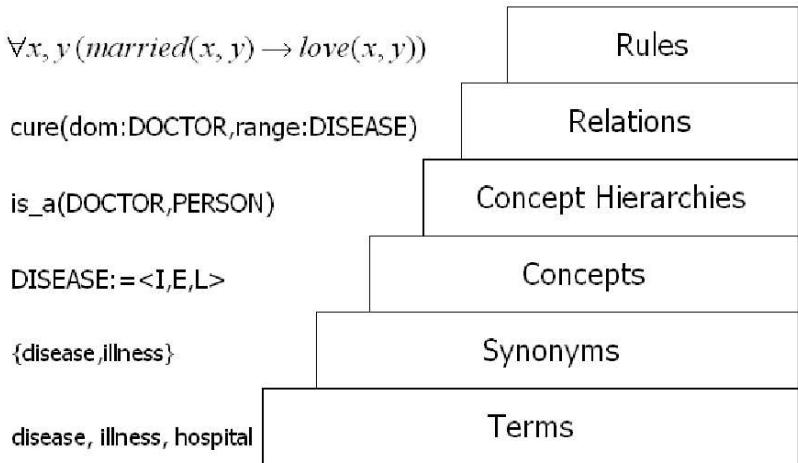


Figure 2 - Layered Representation of Knowledge

The figure is oriented among the concept “disease”. The base line includes terms which belong to a specified domain, and then the synonyms are grouped for the terms that refer to the same meaning. IS-A relation is defined for the conceptual hierarchies. One step further is the relations which defines a particular event like “cure”. This degree can be defined as the concept “doctor” has a “cures” relation with the concept “disease” which represents a single knowledge part. Rules can be defined on top these relations like the relation “married” implies the relation “love” between x and y.

Extraction of semantic concepts and defining their relations from unstructured texts is a complex problem since the human language should be analyzed and converted to a machine readable format without losing the context and inference. There exist more than one way to explain a vulnerability and its mitigation process. Automated approaches should classify these different explanation types and associate them in a common context.

CHAPTER 2

LITERATURE REVIEW

In this section standard and novel approaches within the conducted research are examined and summarized. The survey starts by seeking ontology modeling techniques in general and especially the specialized ones to the Cyber Security domain. Majority of approaches are manually modeled ontologies in which researchers struggle to solve the domain dynamics with or without getting help from security experts. Semi-Automatic and Automatic approaches the focusing target of this study, therefore more emphasize is given to this kind of researches in the literature survey. The other significant topic is *Semantic Concept Extraction* which constitutes the core part of this study especially from the technical perspective. The literature survey for these categories is examined in the following sections in detail.

2.1 Cyber Security Ontology Modeling

One of the major research tasks related to semantic web area is development of ontology for the given domain. There are manual, semi-automatic and automatic approaches for ontology development process. The common purpose of these techniques is to construct an ontology which can be used for further semantic analysis. The ontology development process is also regarded as ontology modeling, since in majority of approaches the task is involved with a manual analysis and modeling processes. Analysis should be deep enough to cover entities which are significant and also stands on top levels from the perspective of taxonomy. To meet with the ontological requirements, manual approaches still mostly preferred ones by the community. The main reason is ability to construct a static querying system on these manually modeled ontologies utilizing semantic querying languages like SPARQL or ARQ. Once the ontology is modeled as an initial step, the mapping processes from external or internal data becomes subsequent tasks to accomplish. Static structure of the model compels these mapping processes to manual modifications. Automatic approaches include dynamic ontology population methodologies which extracts concepts from different sources with their relations. Existing ontologies, domain related free text

sources, documents in structured formats or terminologies in text books can be sources of the automatic approach. Text processing, information retrieval and data mining methodologies are used in conjunction with NLP based methodologies during the construction and mapping phases. On the other hand, semi-automatic approaches that are compound of the manual and automatic processes are evolving rapidly. Previous researches related to each type of ontology development approach are examined in detail in the following sections. Semantic concept extraction which is one the core parts in this work is elaborated in an individual section under the related work title, since the area consists of cross-cutting methodologies for ontology development and can be used for any kind of ontology development process. Variety of extraction approaches are used in singular or hybrid systems which contributes to the ontology construction phase.

2.1.1 Manual Approach

Herzog creates ontology within Information Security Domain [2]. Main ontology entities are Asset, Threat, Vulnerability, Security Goal, Defense Strategy and Countermeasures (OWL file is publicly available). Refinements of these concepts are Human, Technology and Credential. Corresponding non-core concepts are Product, Operating System and Model. Relations are also categorized in two for core and non-core concepts. They started to model with upper layer entities and continued with the refinements of these concepts. The implementation was made manually. Extensions to the refinements may be done as future work.

Vorobie and Bekmamedova [1] defines a case study which is illustrates a Mitnick attack against a Gaming System. Their core point is to provide security for especially distributed and complex software systems. For this reason, they constructed ontology for security threats and countermeasures by using existing ones. The aim is to create a vocabulary basis for security knowledge exchange which is understandable from both humans and machines. Thanks to this common information exchange system, security detection and prevention mechanisms can for collaboratively by sharing event information with each other.

CYBEX is ontology for Cyber Security information exchange. [3] Cyber Security information is examined under three main operational domains which are Asset Management, Incident

Handling and Knowledge Accumulation. Each domain has its own operators, databases and information share mechanisms. CYBEX defines a protocol for exchange cyber security data between these domains and constitutes a bridge for information sharing between different commonly accepted formats like CVE, CPE, CVSS, OVAL, CAPEC and MAEC.

Undercoffer, Joshi and Pinkston defined IDS ontology. [5] They manually analyze over 4000 classes from ICAT (Internet Catalog of Assailable Technologies) maintained by NIST. Taxonomy and Languages are utilized in ontology generation processes. Main entities for IDS ontology are HOST, ATTACK, INPUT and MEANS. In this work, a simple reasoning mechanism is also introduced in order to validate the IDS ontology. Instance hierarchy of the entities provides opportunity to adjust granularity level while performing correlation process.

Nguyen gathers the information system ontologies with respect to their domains [12]. Information Security domain is one of them which includes twelve main ontology directly related and six of them partially related with the topic. All of them are briefly mentioned, however two ontologies which are Herzog's and Fenz's are especially emphasized and example hierarchies for top level threat and countermeasure concepts are also given in the work.

2.1.2 Automatic or Semi-Automatic Approach

M. Balakrishna and M. Srikanth present a semi-automatic development of an ontology library for the 33 topics defined in NIST. [7] They used an external state of the art tool which is called Jaguar - KAT for the knowledge acquisition and domain understanding. The tool performs complex text processing using advanced NLP methodologies including named-entity recognition, part of speech tagging, WSD, co reference resolution and semantic parsing. In addition to these, Knowledge management/classification algorithms are also utilized knowledge accumulation. For each NIST title, a set of documents are collected manually including sentences related to the topic. Rest of the work is performed by the tool Jaguar which iteratively processes sentences n times until the nouns of the sentences are extracted with their "IS-A", "PW" (Part-Whole) and "Cause" relations. Ontology for each topic is generated separately, and then that ontology is merged via a resolution algorithm. JAGUAR Ontology Builder Tool is created by LYMBIA Corporation. [7]

AHMAD, K. and GILLAM introduce an automatic ontology extractor from unstructured texts. [6] The algorithm used for ontology extraction includes only "IS-A" relations. The algorithm is based on the repetition of phrases within text documents. "Weirdness Index" formula is used for concept selection to the ontology. Concepts with ISA relations are substrings of each other which is very straightforward.

Bedini and Nguyen examined different ontology generation methodologies. [8] They specified criteria for automatic ontology generation process and evaluation. Those criteria are Analysis, Generation, Validation, Evolution and Extraction. Ontology generation methodologies are discriminated under different levels of automation which are semi-automatic approach, optional human intervention and completely automatic. "Learning OWL ontologies from free texts" and "LOGS" business models are applicable for my work* which include Text source analysis, NLP engine, morphological and semantic analysis, machine learning approach in extraction, Similarity based on concepts and relationship analysis in analysis phase, different format, internal ontology structure based on lattice in generation phase and validation for each module within validation phase. This work claims that these approaches seem to have not met great consensus within the community***.

SALT is another process to generate domain ontologies from text documents. Three types of knowledge sources are required for this process. A well-defined general reference ontology, an external resource to discover lexical and structural relationships between terms, and a coherent set of training text documents. The process includes NLP approach in order to extract terms from documents with respect to the reference ontology and defines relations between them. To complete the process a manual concept refinement is required as well. Since irrelevant concepts should be eliminated from the sub-ontology. According to this source [8] with a large set of training documents, this methodology achieves really good results.

In the work [9], a bootstrapping approach is proposed for developing a Cyber-Security ontology using textbook index terms. This approach fits to the description of SALT system in [8] in terms of two requirements which are reference ontology, external resource (Security Textbook Index). Wikipedia texts are also used for similarity matching. Bootstrapping security ontology consists of the following processes; Preprocessing the index file (Identifying relationships with static index analysis), Developing Security Ontology from index file (Constructing ontology in protégé with terms and relations extracted previously), Exact string matching (Matching concepts within the reference ontology and the index

terms, by this way enriching the ontology with existing relations coming from reference ontology), Substring matching (Same with previous step additionally with Wikipedia category matching but with the substrings), Prefix-Postfix modifier matching (Start and end modifiers in index terms linked with the ontology), Processing the leftover terms (Remaining index terms are processed again in order to match with existing categories), Importing security concept definitions (Definitions of security concepts are imported from NIST for existing ones). As a result of the work, Herzog's existing ontology which includes 463 concepts was increased to 638 concepts.

In the study [\[10\]](#), authors claim that the ontological approach can be utilized by SIEM (Security Information and Event Management) systems which have huge repositories for various types of security event data including Network configurations security policies, vulnerabilities, arracks, service dependencies alerts etc. These data is accumulated in a huge repository for further event filtering, reasoning, virtualization, decision making and security evaluation purposes. Data is provided by structured cyber security related information sources like CPE, CCE, CVSS and CVE. Whole data is held in relational databases and rarely in xml files. The idea is to support current SIEM systems by providing ontological approach in conjunction with the existing relational database usage. The core of the hybrid system is not explained well, how they work collaboratively and individually remain unclear.

Instead of generating a new ontology from scratch, the authors of [\[11\]](#) proposed a mapping methodology for existing information security knowledge bases which are EBIOS (Represents a method for assessment and treatment of IT security risks created by French Ministry of Defense) and IT Grundschutz Manual (Aspects and applications, threat catalogs and safeguard catalogs collected by German Federal Office for Information Security) to the security ontology. The steps they followed are as follows; Ontology analysis, Knowledge base analysis mapping concepts and relations, mapping the knowledge and the evaluation phase. Top level concepts and corresponding relations are defined for the security ontology. Two existing knowledge bases are analyzed in order to match categories with the predefined top level concepts and relations. After two models are constructed, mapping is performed automatically between the security ontology and knowledge bases. Difficulties faced during this mapping process are also explained within the work.

In the study [\[13\]](#), the authors generate a modeling ontology for integrating vulnerability entities to security requirement concept basics. The initial step is to examine vulnerability catalogs both in general manner and network specific ones. CORAS project is also

mentioned as a vulnerability assessment framework which indicates vulnerability relations. The proposed ontology is defined by an abstract meta-model. The aim in this work was to prove that different security models can be integrated to an ontology which has capability to perform different types of analyses for same concepts.

J. A. Wang and M. Guo defined ontology for managing vulnerabilities. They model top level concepts based on CVE, CWE, CPE and CAPEC with the standard manual ontology modeling methodologies. [\[14\]](#)

The purpose of the work done in [\[15\]](#) is to make a security automation tool via constructing security vulnerability ontology and perform reasoning based on this knowledge base. Their principle source is NVD for the ontology construction phase. One of the significant contributions of their work is the usage of vulnerability descriptions within the CVE structured data. They defined an extended BNF (Backus–Naur Form) by analyzing vulnerability description texts syntactically. They created a conceptual model for their vulnerability ontology which includes the following top level concepts; "Vulnerability", "Introduction_Phase", "Active_Location", "IT_Product", "IT_Vendor". "IT_Product", "IT_Vendor" concepts are extracted from CPE which matches with a specific regular expression form. Other concepts are determined with respect to the taxonomical structure of the vulnerabilities. As a result of this work, they proved the concept of security automation through semantic technologies.

Authors in the work [\[16\]](#) build ontology for cyber security operational information. Their principle aim is to develop an ontological approach for the cyber security in cloud computing domain, to perform this task they initially constructed their ontology focusing non-cloud computing domain and then apply that to the cloud-computing. The proposed ontology of this work includes three domains which are IT-Asset Management, Incident Handling and Knowledge Accumulation domains. Each of them has their own specific database, For IT-Asset management domain; there are Provider Resource DB and User-Resource DB. Incident DB and Warning DB are manipulated by incident handling domain. These databases are used by their corresponding actors within a cyber-security operational context. These data refers to the knowledge bases within the knowledge accumulation domain. These knowledge bases are Version, Configuration, Assessment, Detection/Protection, Vulnerability and Threat knowledge bases. In addition to these, the work indicates the major cyber security information standards like ARF, CVSS, CEE, CAPEC, OVAL, CPE and CCE with respect to the pre-defined domains. In the scope of the second

phase of the work, each domain and corresponding databases are examined from the perspective of the cloud computing principles which are cloud service subscription information, resource dependency information, security level information, data provenance, data placement change log, incident/event information, cloud service enumeration and taxonomy. As a consequence of this analysis they provide the essential difference between cloud and non-cloud systems within the context of cyber security. For instance, data ownership right preservation is dependent on different aspects for both systems. They also claim that, the security paradigm is shifting from protecting systems to the maintenance of the systems without losing availability.

2.2 Semantic Concept Extraction

[17] Is one the researches that concerns with the structured and unstructured raw data conversion to the semantic linked data. The purpose of their work is to collect cyber security related text information from different sources and combine them via their system which analyses them and tries to extract concepts with their links and bound to the defined cyber security ontology. They claim that this ontology can be used automating early vulnerability identification, mitigation and prevention tasks. Their system architecture consists of three main components which are a CRF-based entity-concept spotter which identifies relevant concepts and entities from the given text, second one is an ontology based RDF triple generator that converts the data provided by the spotter to the ontology and the last one is a link generator that utilizes DBpedia Spotlight to link existing concepts within the current ontology and the DBpedia resource pages. The initial ontology modeling step is performed manually based on the NVD schema. Top level concepts are Vulnerability, Product, Attack and Weakness. The main reason why they manually model the ontology and top level concepts is the deep nesting and similarity in naming conventions of NVD schema. Undercoffer's cyber security ontology [5] is used for reference vocabulary repository. After the ontology is modeled, NVD concrete data is used to populate the ontology, this process is also performed manually since there is a different schema between NVD and their proposed ontology. The proposed extractor is trained using the Stanford NER which utilizes a CRF based NER framework. The training dataset consists of 30 security blogs, 240 CVE descriptions and 80 official security bulletins from Microsoft and Adobe. This corpus is manually annotated by SMEs (Subject Matter Experts). At this point an additional

step for the vulnerability descriptions is also included during the concept extraction phase. Free texts within descriptions are annotated via DBpedia Spotlight which a tool for finding direct mappings between is given text and DBpedia resources. No directly mapped entities are also annotated based on a confidence value. By using this approach they strengthen their cyber security ontology by improving relations with DBpedia. Consequently, they generate a linked data repository for security practitioners and system administrators.

The proposed study in [18] provides an automated extractor for vulnerability information which can be useful especially in Home computer security domain. Extractor systems consists of two different approaches which are machine learning based solution and an NLP based solution. 210 randomly selected vulnerability descriptions constitute the corpus of the work. For the first machine learning driven approach, feature selection and model training is performed, for the NLP approach POS tagging technique is applied to the vulnerability descriptions. As a result of the manual examination of the vulnerability descriptions, they proposed an algorithm for extracting related phrases. To support the NLP approach, SentiWordNet is used to differentiate negativity or positivity within the phrase, by using that attacker and victim actors can be separated more clearly. For the evaluation phase, they utilized from Joshi Corpus and NVD corpus. As a consequence, NLP approach is more successful than machine learning approach.

In the study [19] a prototype system is proposed to extract information about vulnerabilities and attacks from the web sources. The system utilizes Wikitology which is a general purpose knowledge base derived from Wikipedia to extract related concepts an DBpedia which is used in mapping of extracted concepts to resources. The system is developed on the NVD. The system collects data about streaming security information between different sources. These sources can be categorized as blogs, technical news, discussions on forums and chat-room, text descriptions relevant and irrelevant. These data is collected and classified with an SVM Classifier. Resulting vulnerability descriptions are analyzed for concept extraction based on the Wikitology source and the computer security taxonomy in order to extract vulnerability terms which are asserted to the common knowledge base by security experts manually. The aim is convert the raw data to the machine understandable format in order to ease the burden of security experts which consequently decides the assertions to the knowledge base. For the evaluation phase, for each text description which is mainly taken from vulnerability descriptions are examined by authors who have a certain background about the security domain. As a result of these

manual examinations, top n concepts are selected and ranked. Then the results of the automated system is compared with the manual annotations and it is concluded that the success rate is around 89.47% for the existence accuracy of concepts and 80% for the ranked order comparison.

Authors in [20] propose an automatic labeling system for entity extraction within cyber security domain. Data sources for automatic labeling process is based on the NVD text description fields, CVE-ID, Microsoft security bulletins and Metasploit Framework which is a database of available exploits that includes a text description, categorization and properties that reference to the vulnerabilities in NVD. Auto-Labeling process includes database matching and heuristic rules. In order to represent vulnerability, a corresponding gazetteer of relevant terms is constructed. In addition to these, a distinctive approach is applied which is for entity extraction via sequential labeling within the scope of this work. This approach is based on sequential tagging models which are solved with the Maximum Entropy Models and with the support of mathematical theories. The main reason why these models are used is to perform the task of the POS tagging systems. The concept that is desired to prove within this work is to outperform POS tagging with this proposed system. As a result of the work done, the unstructured text data can be annotated via utilizing from existing cyber security related structured data and the developed automatic tagging system.

CHAPTER 3

METHODOLOGY

The methodology applied for the proposed solution is explained in this chapter. Design considerations and general system processes are mentioned in the overview and system architecture sections. The rest of the sections are dedicated to the each module within the system. The novel technique applications are emphasized in these module explanations, the reused and utilized technologies are not explained in detail. For interested people with these used technologies, extra sections are attached to the appendix section of the study.

3.1 Overview

In this section, the major parts of the proposed solution are indicated in most general format. These major parts are the modules that constitute the overall system. In figure-3, the big picture for the system is given. Comprehensive information and core details will be given within the following sections for each module.

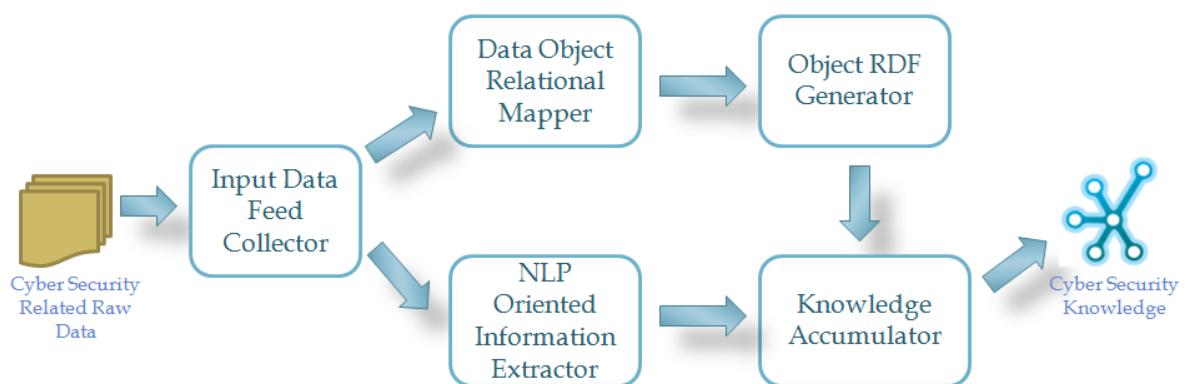


Figure 3 - General Block Diagram of the System

The figure-3 above represents the input, core processor parts and the output of the whole system in a very generic point of view. The initial process has been started with the

acquisition of Cyber Security related raw data by the input feed collector module. This module is responsible of both the automatic feed fetching from the static web services and also processing the fetched data by differentiating it to the structured and unstructured parts since they are evaluated by different modules subsequently. The Data Object Relational Mapping module takes the structured part of refined data which is in the XML format and converts it automatically to the relational objects which is the desired input format of the Object RDF Generator module. This module takes any relational object tree as and generates triples in RDF format from them in a holistic way. In parallel to this process, NLP Oriented Information Extractor module takes the unstructured part of the data which cannot be transformed directly to any format. This module is indicated as whole for brevity in this generic diagram, however it is compound of different processes which proceeds sequentially and extract RDF triples from the given free formatted text inputs. The generated RDF triples of these two modules are accumulated in a defined ontology which is the final output of this study, a knowledge base for cyber security domain.

3.2 System Architecture

The system architecture is designed based on modularity principle which dictates eliminating dependencies and forces to decouple processes. The aim for applying this principle is to increase the reusability of the modules with different data types, replacement of blocks for utilizing different combination of methodologies and increase the measurability and testing capabilities.

The initial point for the whole process is the selection and preparation of the input which is the combination of structured and unstructured data. This data is collected properly from certain sources and transferred to the object relational mapping module which is an intermediary process to convert raw data into relational objects that are suitable for preceding modules. Object RDF generator and NLP oriented information extractor function in parallel, both modules uses different parts of the same data title and produces homogenous output which feeds the knowledge accumulator module. Targeting knowledge base is constructed by the sequential work of these modules.

The figure-4 below also indicates the overall processes of the system including relations of the modules and static content representations. On the left hand side, there are commonly

accepted standards for cyber security community. This is the semi-structured data which is collected and regulated to the desired input format for the object mapping process. In the current scope of the project, mapper module takes XML formatted data inputs only. The main task of this module is to convert given XML data to the corresponding relational objects. Relational objects are the principle source for analysis and interpretation of the structured content. The flow of the data continues with two parallel processes which are NLP oriented knowledge extractor and relational object knowledge extractor. As inferred from the name, the output of the both modules is the common knowledge. However, the input data format and inner working principles for these modules are completely different.

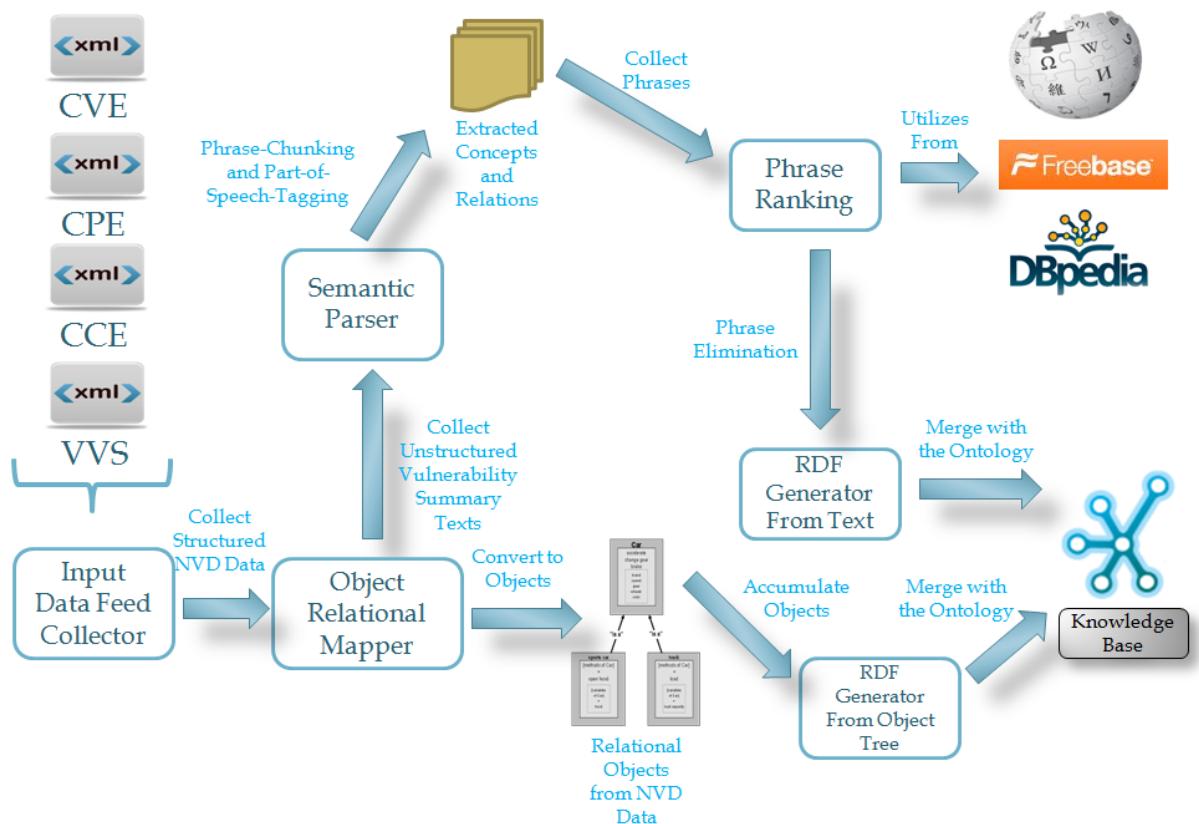


Figure 4 - Overall Process Diagram of the System

3.3 Automatic Data Feed Collector

The purpose of the system is essentially converting raw data to knowledge. Therefore this module stands for a significant role for both fetching and collecting external data which are the source of other modules. At this point, raw data does not mean any particular kind of data; however it refers to both structured and unstructured data. In this work, candidates for input data collection are analyzed and selection is performed carefully, since the quality of the system output, success rate and overall throughput of the system is highly dependent on this selected data.

3.3.1 Cyber Security Related Data Analysis

In the first phase of the cyber security related data selection, structured data standards within security domain is elaborated. As the concept of security gained importance in cyber space, the need for structured cyber security related data has been arisen. Contemporarily, there are certain organizations which support the systematically collection and maintenance of this data. The aim here is to standardize representation of Vulnerabilities, Threats, Countermeasures, Products and their breakdowns with a complete hierarchy. The initial point for pre-research is these organizations and their standards. In the following table-1, organizations are given with their corresponding data standards.

MITRE
CRF (Common Result Format)
CEE (Common Event Expression)
CWSS (Common Weakness Scoring System)
CybOX (Cyber Observable Expression)
TAXII (Trusted Automated eXchange of Indicator Information)
NIST
ARF (Asset Reporting Format)
CCE (Common Configuration Enumeration)
CCSS (Common Configuration Scoring System)
CPE (Common Product Enumeration)
OCIL (Open Checklist Interactive Language)
ITU-T
CVE (Common Vulnerability and Exposures)

CVSS (Common Vulnerability Scoring System)
CAPEC (Common Attack Pattern Enumeration and Classification)
CWE (Common Weakness Enumeration)
MAEC (Malware Attribute Enumeration and Characterization)
OVAL (Open Vulnerability and Assessment Language)

Table 1 - Cyber Security Information Sources with Organizations

MITRE is a corporation with the aim of providing a common approach which allows for a collective response to cyber security threats. The focus is to expand the use of common terminology and structures.

ITU-T is Telecommunication Standardization Sector has a division for security data standardization which conducts projects for different layers of security. There are ongoing projects for creating security guidance manuals, security assessment and evaluation criteria, network security, risk Assessment etc.

NIST stands for the National Institute of Standards and Technology; it is conducted by U.S. Department of Commerce. NIST not only includes standards for cyber-security field but also in different subject areas like bioscience & health, energy, nanotechnology etc.

NVD is the National Vulnerability Database which is supported by NIST and specified to the area of cyber security. NVD includes the most widespread security databases which are represented using the SCAP (Security Content Automation Protocol) format. NVD accumulates the security related information and maintains within the corresponding databases in standard format. Primary resources maintained within the scope of NVD are the following; CVE, CCE, XCCDF, OVAL, CPE, CVSS, CWE. All these databases are considered as SCAP data feeds which are publicly available by NVD data services. In this research, each of these data feed content is analyzed in detail in terms of usability among the system.

The selection process is performed with respect to the variety of the items and their relations within the content. Degree of relations contributes to the knowledge base structure and increases the quality of the constructed ontology; terms are also significant for the general coverage of the ontology which also can be utilized as a term dictionary. Selected data formats and selection reasons are explained in the following sections in detail.

3.3.1.1 Vulnerability Feeds

This data includes security related software flaws. Each flaw is enumerated with a global identifier with the prefix of "CVE-'Year'". For each vulnerability, there exists a published and modified time. Source(s) and reference(s) associated with the vulnerability are also placed within the xml. These relations can be free text or a web link as well. Vulnerable product configurations and weakness categorizations are also indicated in a vulnerability entry. An example for a structured vulnerability data taken from NVD data feeds is illustrated in the figure-5. NVD also provides XSD schema for the structured vulnerabilities which is the key factor for selection of this data for the input collection.

```
<entry id="CVE-2013-6737">
  <vuln:vulnerable-configuration id="http://www.nist.gov/">
    <cpe-lang:logical-test negate="false" operator="AND">
      <cpe-lang:logical-test negate="false" operator="OR">
        <cpe-lang:fact-ref name="cpe:/a:ibm:storwize_unified_v7000_software:1.4.2.0"/>
        <cpe-lang:fact-ref name="cpe:/a:ibm:storwize_unified_v7000_software:1.4.1.1"/>
      </cpe-lang:logical-test>
    </cpe-lang:logical-test>
  </vuln:vulnerable-configuration>
  <vuln:vulnerable-software-list>
    <vuln:product>cpe:/a:ibm:storwize_unified_v7000_software:1.4.1.0</vuln:product>
  </vuln:vulnerable-software-list>
  <vuln:cve-id>CVE-2013-6737</vuln:cve-id>
  <vuln:published-datetime>2014-06-21T11:55:03.540-04:00</vuln:published-datetime>
  <vuln:last-modified-datetime>2014-06-23T11:49:09.890-04:00</vuln:last-modified-datetime>
  <vuln:cvss>
    <cvss:base_metrics>
      <cvss:score>4.0</cvss:score>
      <cvss:access-vector>NETWORK</cvss:access-vector>
      <cvss:access-complexity>LOW</cvss:access-complexity>
      <cvss:authentication>SINGLE_INSTANCE</cvss:authentication>
      <cvss:confidentiality-impact>PARTIAL</cvss:confidentiality-impact>
      <cvss:integrity-impact>NONE</cvss:integrity-impact>
      <cvss:availability-impact>NONE</cvss:availability-impact>
      <cvss:source>http://nvd.nist.gov</cvss:source>
      <cvss:generated-on-datetime>2014-06-23T11:49:09.657-04:00</cvss:generated-on-datetime>
    </cvss:base_metrics>
  </vuln:cvss>
  <vuln:cwe id="CWE-264"/>
  <vuln:references xml:lang="en" reference_type="VENDOR_ADVISORY">
    <vuln:source>CONFIRM</vuln:source>
    <vuln:reference href="http://www.ibm.com/support/docview.wss?uid=ssg1S1004676" xml:lang="en">http://www.ibm.com/support/docview.wss?uid=ssg1S1004676</vuln:reference>
  </vuln:references>
  <vuln:summary>IBM System Storage Storwize V7000 Unified 1.3.x and 1.4.x before 1.4.3.0 does not properly restrict the content of a dump file upon encountering a 1691 hardware fault</vuln:summary>
</entry>
```

Figure 5 - Example CVE Data Feed

CVSS is another provided format aiming to standardize a common vulnerability scoring system. It is an open framework for identifying the characteristics of IT vulnerability. It has Base, Temporal and Environmental groups each of them produces a numeric score ranging

from 0 and 10 [41]. After the detailed analyses of the data, it can be seen clearly that the weight of base metrics is very high compared to other metrics. Base metrics consist of Access Vector, Access Complexity, Authentication, Confidentiality Impact, Integrity Impact and Availability Impact. These metrics are represented with predefined enumeration values, for instance Access Vector metric can be represented with the following enumeration values; Local (L), Adjacent Network (A) and Network (N). These values are used to indicate a property of an entity's relation, however they are exist in free texts and used by the definition of threats or vulnerabilities. There can be found many sentences in cyber security blogs including word "Adjacent Network". Therefore, they can be mapped to the extracted concepts from free texts with exact matching techniques which contribute to the development of the linked data as well. On the other hand, for the Access Complexity (AC) measurement, this theory is not applicable and will not promise good results for linked data development since the metric values for this measurement is denoted as "High", "Medium" and "Low" which is hard to link to the other concepts in within a reasonable scope. These kinds of enumerations are only useful for similarity matching within a controlled and limited domain.

One of the most significant parts for CVE data is the description field provided by vulnerability entries. It seems a free formatted human written text, however with a deep examination, it can be seen that the description paragraphs include consistent structure in terms of syntactically and also usage of terms. This is the reason why description fields of the CVE data is selected as the prior input for the linked data extraction from unstructured texts processes within the NLP Oriented Knowledge Extractor module. Details related to this issue will be given on the individual section of this module.

3.3.1.2 Vulnerability Vendor Statements

NVD provides a platform for software development organizations which enables them to submit their own statements related to the vulnerabilities under the name of "Official Vendor Statements". Organizations can utilize from this service by contacting NVD staff. The structure of this data is organized as follows; the statement entity in which the actual comment is placed in free text format; name of the corresponding CVE, last modified date, contributor which includes generally the name of the staff that writes the statement and the name of the organization. The manual analyses of these statements indicate that these

texts are useful in knowledge base. Statements are including already an existing explicit relation between CVE and Vendor Statements which is the reference point for knowledge generation. In addition to this used terms (Products, Vulnerabilities and Threats) are common with the existing NVD data which enables the augmentation process within the system via NLP Oriented Knowledge Extractor module. As a result VVS can be used for both structured and unstructured data input and takes its place in the data collector module. An example for the VVS XML data is given in the figure-6 below;

```

<nvd:statement contributor="Joshua Bressers" cvename="CVE-1999-0523" lastmodified="2007-09-11T00:00:00.000-04:00"
organization="Red Hat">Red Hat Enterprise Linux by default does respond to ICMP echo requests
, although it's likely that in a production environment those would be filtered by some firewall on entry to your network.
However you can happily block ICMP ping responses using iptables if you so wish, but note that there is no known vulnerability in allowing them.
For more details, please see:
http://kbbase.redhat.com/faq/FAQ\_43\_4304.shtml</nvd:statement>
<nvd:statement contributor="Joshua Bressers" cvename="CVE-1999-0524" lastmodified="2010-01-05T00:00:00.000-05:00" organization="Red Hat">Red Hat Enterprise Linux is configured by def
<nvd:statement contributor="Joshua Bressers" cvename="CVE-1999-0997" lastmodified="2006-09-27T00:00:00.000-04:00" organization="Red Hat">Red Hat does not consider CVE-1999-0997 to be
<nvd:statement contributor="Mark J Cox" cvename="CVE-1999-1199" lastmodified="2008-07-02T00:00:00.000-04:00" organization="Apache">Fixed in Apache HTTP Server 1.3.2:
http://httpd.apache.org/security/vulnerabilities\_13.html</nvd:statement>
<nvd:statement contributor="Seapine Software" cvename="CVE-1999-1567" lastmodified="2010-07-22T00:00:00.000-04:00" organization="Seapine Software">This issue was originally reported
<nvd:statement contributor="Mark J Cox" cvename="CVE-1999-1572" lastmodified="2007-03-14T00:00:00.000-04:00" organization="Red Hat">Red Hat Enterprise Linux 5 is not vulnerable to th
<nvd:statement contributor="Mark J Cox" cvename="CVE-2000-0505" lastmodified="2008-07-02T00:00:00.000-04:00" organization="Apache">Fixed in Apache HTTP Server 1.3.14:
http://httpd.apache.org/security/vulnerabilities\_13.html</nvd:statement>
<nvd:statement contributor="Al Menendez" cvename="CVE-2000-0572" lastmodified="2007-02-22T00:00:00.000-05:00" organization="Razor">Subsequent releases of Razor address this issue and
|
```

Some additional notes ...

- With version 4.1 and above, administrators of Razor may switch and use the local OS authentication instead of Razor's authentication method.
- OS permissions and protections always apply to the artifacts stored in the database.
- This notice applies to users that have already logged into the supporting system. This primary means of defense is intact inspite of this particular vulnerability.
- The next Razor release (due out in mid-2007) will allow remote UNIX clients to utilize SSH to authenticate the remote user. More information on this release and others may be found

<http://www.visible.com/Products/Razor>

Please contact Visible Systems Corporation at 1-800-6-VISIBLE if you have additional questions.</nvd:statement>

Figure 6 - Example VVS Data Feed

3.3.1.3 Common Product Enumeration

CPE is the repository of products which are target of a threat or has vulnerability stated previously in NVD. Enumeration of products can be reach from other external data providers. However the structure of the product data is very primitive which does not require any extra analysis process or association efforts. The long name and the abbreviations of the products are extracted from the CPE data feed and utilized with this raw format. Therefore this data can be regarded as a product dictionary which can be used for general purpose. In the scope of this work, product enumerations are used for explicitly linking with existing vulnerability data. There exists an xml schema for different versions which includes detailed breakdowns for the data; however the actual data feeds does not

include these detailed relations about products. For this reason those XSD schemas are discarded in this work and the only property used within data is the name of the CPE. The enumeration for the name represents a common format. This format is parsed with a regular expression. The format information is taken from MITRE Corporation CPE Specifications Archive [Reference needed]. For CPE v2.3 representation of a sample product named “Microsoft Internet Explorer 8.0.6001 Beta” is as follows;

```
wfn:[part="a",vendor="microsoft",product="internet_explorer",
version="8\.0\.6001",update="beta"]
```

This format is converted to the following final form;

```
cpe:/a:microsoft:internet_explorer:8.0.6001:beta
```

The whole data is published in this format by NVD. At this point the CPE converter process functions in the Input Collector module. It takes the CPE entry and converts it to the following XML format which is the desired XML format for unstructured data processing modules of the system;

```
<cpe>
  <Vendor>Microsoft</Vendor>
  <Product>internet_explorer</Product>
  <Version>8\.0\.6001</Version>
  <Update>Beta</Update>
</cpe>
```

Figure 7 - Example CPE Data Feed

An extra step is required for modification of the corresponding XSD structure of the CPE which is relatively simple compared to the other data sources.

3.3.1.4 Common Configuration Enumeration

Configurations are the focus points of systems. They are the most delinquent parts that cause conspicuous vulnerabilities for especially software systems. The more software

corporations develop flexible systems, the bigger configuration files will get and this increases the customizability of the system as well as its complexity. At this point IT personnel should be very careful when integrating new software to the system or updating existing software. Configuration issues arouse which should make manually and requires a consistent concentration. Information about configurations are assumed to include within the software package, however these information can be highly distributed and also controversial with other versions of the same software.

The purpose of CCE is to enumerate configuration data across multiple information sources and software tools. This data includes detailed information about a particular configuration issue. Each entry represents a configuration item with its published date, modified date and a text summary in which the issue is stated. This statement includes the target of the configuration as well as the corresponding task to perform for this configuration issue. This free text is one the valuable inputs for linked data extraction. It can be used in NLP oriented data extractor module.

```

<entry id="CCE-5148-2">
  <config:cce-id>CCE-5148-2</config:cce-id>
  <config:published-datetime>2009-07-30T19:31:11.670Z</config:published-datetime>
  <config:last-modified-datetime>2012-05-25T03:56:47.663Z</config:last-modified-datetime>
  <config:summary>The "Refuse machine account password change" policy should be set correctly.</config:summary>
  <scap-core:control-mappings>
    <scap-core:control-mapping system-id="http://csrc.nist.gov/publications/PubsSPs.html#SP-800-53-Rev.8203"
      source="http://nvd.nist.gov/" last-modified="2009-08-20T16:15:59.360Z">
      <scap-core:mapping published="2009-08-20T16:15:59.360Z">AC-3</scap-core:mapping>
      <scap-core:mapping published="2009-08-20T16:15:59.343Z">CM-6</scap-core:mapping>
      <scap-core:mapping published="2009-08-20T16:15:59.343Z">CM-7</scap-core:mapping>
      <scap-core:mapping published="2009-08-20T16:15:59.343Z">SC-5</scap-core:mapping>
    </scap-core:control-mapping>
  </scap-core:control-mappings>
</entry>
<entry id="CCE-5045-0">
  <config:cce-id>CCE-5045-0</config:cce-id>
  <config:published-datetime>2009-07-30T19:31:11.703Z</config:published-datetime>
  <config:last-modified-datetime>2009-07-30T19:31:11.703Z</config:last-modified-datetime>
  <config:summary>The encryption algorithm to be used by EFS should be properly chosen.</config:summary>
  <scap-core:control-mappings>
    <scap-core:control-mapping system-id="http://csrc.nist.gov/publications/PubsSPs.html#SP-800-53-Rev.8203"
      source="http://nvd.nist.gov/" last-modified="2009-08-20T16:16:00.577Z">
      <scap-core:mapping published="2009-08-20T16:16:00.577Z">CM-6</scap-core:mapping>
      <scap-core:mapping published="2009-08-20T16:16:00.577Z">CM-7</scap-core:mapping>
      <scap-core:mapping published="2009-08-20T16:16:00.577Z">SC-2</scap-core:mapping>
    </scap-core:control-mapping>
  </scap-core:control-mappings>
</entry>
<entry id="CCE-4736-5">
  <config:cce-id>CCE-4736-5</config:cce-id>
  <config:published-datetime>2009-07-30T19:31:11.733Z</config:published-datetime>
  <config:last-modified-datetime>2009-07-30T19:31:11.733Z</config:last-modified-datetime>
  <config:summary>The TCPMaxPortsExhausted setting should be properly configured.</config:summary>
  <scap-core:control-mappings>
    <scap-core:control-mapping system-id="http://csrc.nist.gov/publications/PubsSPs.html#SP-800-53-Rev.8203"
      source="http://nvd.nist.gov/" last-modified="2009-08-20T16:14:51.327Z">
      <scap-core:mapping published="2009-08-20T16:14:51.313Z">AC-4</scap-core:mapping>
      <scap-core:mapping published="2009-08-20T16:14:51.313Z">SC-5</scap-core:mapping>
      <scap-core:mapping published="2009-08-20T16:14:51.327Z">SC-7</scap-core:mapping>
    </scap-core:control-mapping>
  </scap-core:control-mappings>
</entry>

```

Figure 8 - Example CCE Data Feed

In addition to the basic properties, a CCE entry includes an inner mapping information with security controls and associated assessment procedures defined in NIST SP 800-53 [26]. Each entry is associated one or more control-mapping entry. These entries are including the corresponding "system-id" which is usually a web address link, the source which is NIST in most cases and the control identifier like "CM-6", "SC-7". Identifiers for controls are listed in NIST Special Publication 800-53 (Rev. 4) [26]. For instance "CM-6" stands for the configuration settings which reside in the family of "CM" (Configuration Management). There is a detailed control description as well as control enhancements provided in NVD and NIST web pages. In the figures 9 and 10 below;

NIST Special Publication 800-53 (Rev. 4)

Security Controls and Assessment Procedures for Federal Information Systems and Organizations

Configuration Management Control Family

Showing 11 controls:

No.	Control	Priority	Low	Moderate	High
CM-1	CONFIGURATION MANAGEMENT POLICY AND PROCEDURES	P1	CM-1	CM-1	CM-1
CM-2	BASELINE CONFIGURATION	P1	CM-2	CM-2 (1) (3) (7)	CM-2 (1) (2) (3) (7)
CM-3	CONFIGURATION CHANGE CONTROL	P1		CM-3 (2)	CM-3 (1) (2)
CM-4	SECURITY IMPACT ANALYSIS	P2	CM-4	CM-4	CM-4 (1)
CM-5	ACCESS RESTRICTIONS FOR CHANGE	P1		CM-5	CM-5 (1) (2) (3)
CM-6	CONFIGURATION SETTINGS	P1	CM-6	CM-6	CM-6 (1) (2)
CM-7	LEAST FUNCTIONALITY	P1	CM-7	CM-7 (1) (2) (4)	CM-7 (1) (2) (5)
CM-8	INFORMATION SYSTEM COMPONENT INVENTORY	P1	CM-8	CM-8 (1) (3) (5)	CM-8 (1) (2) (3) (4) (5)
CM-9	CONFIGURATION MANAGEMENT PLAN	P1		CM-9	CM-9
CM-10	SOFTWARE USAGE RESTRICTIONS	P2	CM-10	CM-10	CM-10
CM-11	USER-INSTALLED SOFTWARE	P1	CM-11	CM-11	CM-11

Figure 9 - Configuration Management Control Family

NIST Special Publication 800-53 (Rev. 4)

Security Controls and Assessment Procedures for Federal Information Systems and Organizations

[All Controls](#) > CM > CM-6

CM-6 - CONFIGURATION SETTINGS

Family:	CM - CONFIGURATION MANAGEMENT								
Priority:	P1 - Implement P1 security controls first.								
Baseline Allocation:	<table border="1"><tr><th>Low</th><th>Moderate</th><th>High</th></tr><tr><td>CM-6</td><td>CM-6</td><td>CM-6 (1) (2)</td></tr></table>			Low	Moderate	High	CM-6	CM-6	CM-6 (1) (2)
Low	Moderate	High							
CM-6	CM-6	CM-6 (1) (2)							
			Jump To: Revision 4 Statements Control Description Supplemental Guidance References						

Control Description

The organization:

- Establishes and documents configuration settings for information technology products employed within the information system using [Assignment: organization-defined security configuration checklists] that reflect the most restrictive mode consistent with operational requirements;
- Implements the configuration settings;
- Identifies, documents, and approves any deviations from established configuration settings for [Assignment: organization-defined information system components] based on [Assignment: organization-defined operational requirements]; and
- Monitors and controls changes to the configuration settings in accordance with organizational policies and procedures.

Supplemental Guidance

Configuration settings are the set of parameters that can be changed in hardware, software, or firmware components of the information system that affect the security posture and/or functionality of the system. Information technology products for which security-related configuration settings can be defined include, for example, mainframe computers, servers (e.g., database, electronic mail, authentication, web, proxy, file, domain name), workstations, input/output devices (e.g., scanners, copiers, and printers), network components (e.g., firewalls, routers, gateways,

Figure 10 - Configuration Settings for CM-6

Control mappings give extra insight about a particular configuration issue to the security experts or any interested person. However this process should be performed manually which requires the following steps; narrowing the data by eliminating none interested configurations which is a hard process since there is a primitive product type based classification mechanism is provided by NVD which does not satisfy the needs at this point. After the data is filtered, manual examination of issues is necessary to find a specified configuration issue. Scanning CCE XML data is not enough for this phase, since the control

mappings are only appeared with their reference identifiers, therefore for each reference the corresponding detailed information should be reached and examined too.

In the scope of data collector module, configuration enumerations constitutes an important part, since improperly made configuration settings expose numerous vulnerabilities within the software systems which cannot be predictable without manipulated by cyber criminals. These enumerations accommodate significant knowledge with a range of primitive configuration adjustments to the selection of encryption algorithm types. This is the main source for security experts to read these sentences and take warnings into consideration. The proposed knowledge base of this study increases the coverage by incorporation of this data.

3.3.2 Input Collection and Differentiation

In the light of the previous sections about data examination, the implementation of the actual collector and differentiator is revealed. As mentioned earlier, selected data feeds are publicly available on the web. The corresponding XSD structures are collected from the web with an additional research. There are different kinds of data fetching approaches exist which are categorized under "Working with Remote Content" title. With respect to the type of data, fetching mechanisms can be performed with XML, Text, JSON and RSS Feeds. Luckily, NIST organization exposes these data feeds with a structured format by labeling each context with the update date, download type, size and corresponding XSD schema version as it is indicated in figure-11. The collector module utilizes from this structure and designated accordingly. Feeds belong to the previous years are collected initially. The recent data refers to the data which are modified within the previous eight days and the modified data includes all modified entries within the current year. The refreshing time interval is stated as approximately every two hours in the NVD homepage.

Feed	Updated	Version 2.0			Version 1.2		
		NVD XML 2.0 Schema NVD XML 2.0 Change Log		Download	Size (MB)	NVD XML 1.2 Schema	
		GZ (https)	ZIP (https)			GZ	ZIP
Modified	1/24/2015 1:03:30 AM	GZ (https)	0.17	GZ	0.10	ZIP (https)	0.10
Recent	1/24/2015 1:01:11 AM	GZ (https)	0.05	GZ	0.04	ZIP (https)	0.04
2002	1/10/2015 6:22:28 AM	GZ (https)	1.41	GZ	1.11	ZIP (https)	1.41
2003	1/22/2015 12:24:03 PM	GZ (https)	0.42	GZ	0.32	ZIP (https)	0.42
2004	1/22/2015 11:57:13 AM	GZ (https)	0.85	GZ	0.62	ZIP (https)	0.85
2005	1/10/2015 5:07:34 AM	GZ (https)	1.34	GZ	0.99	ZIP (https)	1.34
2006	11/14/2014 8:01:24 AM	GZ (https)	2.13	GZ	1.66	ZIP (https)	2.13
2007	1/22/2015 11:04:11 AM	GZ (https)	2.07	GZ	1.59	ZIP (https)	2.07
2008	1/11/2015 4:08:46 AM	GZ (https)	2.22	GZ	1.56	ZIP (https)	2.22
2009	1/18/2015 4:38:28 AM	GZ (https)	2.13	GZ	1.30	ZIP (https)	2.13
2010	1/22/2015 10:29:36 AM	GZ (https)	2.86	GZ	1.34	ZIP (https)	2.86
2011	1/22/2015 10:06:02 AM	GZ (https)	6.27	GZ	3.00	ZIP (https)	6.27
2012	1/22/2015 9:45:11 AM	GZ (https)	2.45	GZ	1.23	ZIP (https)	2.45
2013	1/23/2015 3:43:46 AM	GZ (https)	2.60	GZ	1.26	ZIP (https)	2.60
2014	1/24/2015 3:22:15 AM	GZ (https)	2.17	GZ	1.24	ZIP (https)	2.17
2015	1/24/2015 3:00:33 AM	GZ (https)	0.03	GZ	0.02	ZIP (https)	0.03

Figure 11 - CVE XML Data Feeds with XSD Schemas

The collector performs http requests to the recent and modified data feed web service provided by NVD with a configurable time interval (hourly, daily, weekly or monthly). This synchronous activity is performed regularly which is called "Polling" in software terminology. This step can be performed perfectly for Vulnerability Data Feeds; however the other selected source does not possess this degree of maturity in terms of both structural and agility of modifications. Corresponding XSD files are held with the fetched data in dictionaries within the module. This process is designated as a source independent software process which provides also possibility to use different data source with properly defining the source endpoints or web services. Therefore by only adding new external service resolver parts, the system will gain a new data fetching capability without disrupting the existing structure. The process performed within the first phase of this module is represented in the following figure-12;

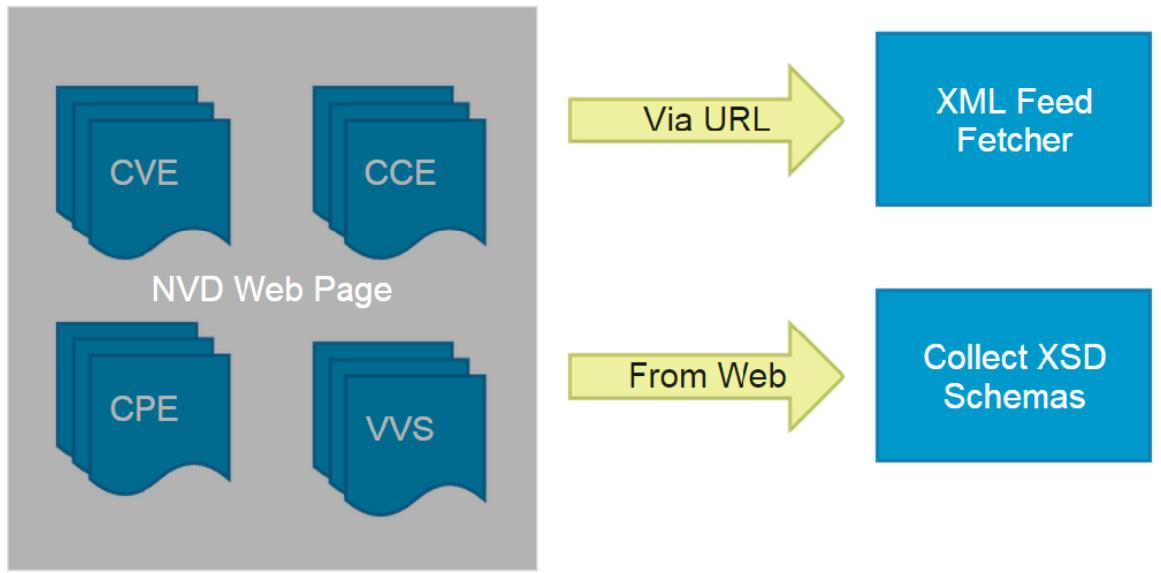


Figure 12 - Block Diagram for Input Collector Module

In the second phase of this module, differentiation process is applied to the fetched raw data. At this point some static operational logic should be added with respect to the coming feedback from data analysis sections. As mentioned earlier, CPE data enhancement should be performed in order not to exceed the boundaries of XML structure. These kinds of processes are performed in this phase and there are two such processes are developed in the scope of this work. These are called CPE enhancement and CCE enhancement processes which are explained in detail in their corresponding CCE and CPE sections.

The output of the second phase is an enhanced, homogenous XML data structure with corresponding XSD data. This structured data source is ready for division and the preceding unstructured data separation phase. Unstructured data can be reached from any external sources which is a simple process, however with the analysis of the selected data it is concluded that usage of uncontrolled raw data does not contribute to the knowledge extraction in a positive manner. It may cause vagueness of data as well as the decreases the quality of the ontology by lowering the precision of the querying processes which is one of the main purposes of the developed ontology. Therefore, unstructured data is selected among the structured XML data which are indicated by the specified tags. Summary and description tags are these selected tags. The module scans the XML data and separates the free texts inside those tags by giving each of them a globally unique identifier (GUID) and name of the tag. This separated data will be used in NLP Oriented Extractor module without

losing the identifier. At this point, maintaining the identifier is very significant, since at the end of the whole processes, identifiers will be the boundary items to accumulate the knowledge. Otherwise, the output of the extraction processes cannot be bound to the existing linked data graph and resides as a different ontology which requires an extra step for merging different ontologies. In order to avoid those extra processes, the problem is solved at the initial phase.

3.4 Structured Data To Relational Object Mapping

As software community shifts to the "Object-Oriented Paradigm", the majority of the software are designed and implemented with the Object-Oriented approach. The main reason for this shift is the approach's power of abstraction on real world objects and relations. This approach also lays the foundations of this module since entities within cyber security domain are also objects which possess different attributes including references to each other. Although processing an XML file seems a straightforward task, when the number of XSD's increase, dynamic data structures and cascading naming conventions are involved within the process, this task becomes more complex and less maintainable. In addition to these, development a pure XML based system is so exhausting and not realistic for maintenance. This is the reason why there exist numerous ORM frameworks which is designated to improve both development speed of the software and increase the degree of understandability. These are the factors that urge to develop a structured data to relation object mapping module. The process diagram for this module is given below figure-13;

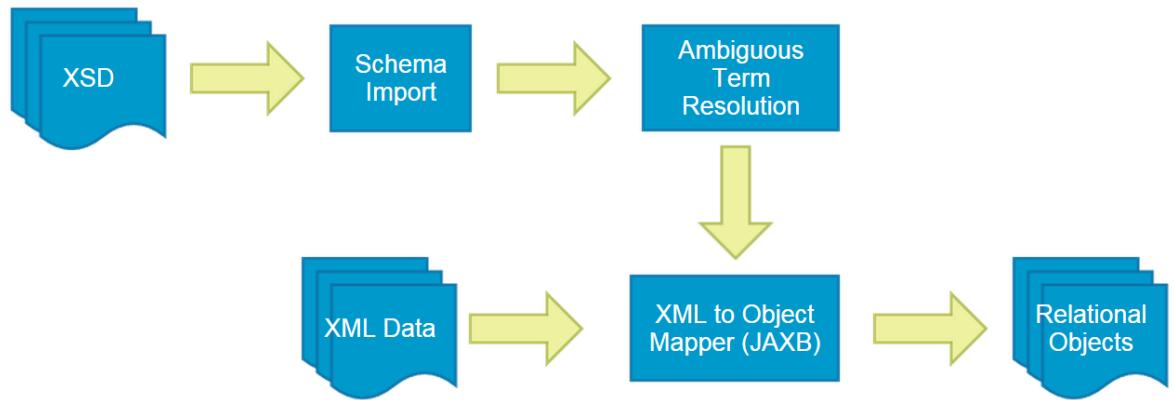


Figure 13 - Block Diagram for Object Relational Mapper Module

3.4.1 XML To Object Tree Conversion

The input data for the module is XML data and corresponding XSD schemas are processed in the previous automatic data collector module. Since the JAVA based technologies are used for software development of this work, a JAVA based framework should be selected for the mapping task. At this point, JAXB 2.2 (Java Architecture for XML Binding) which is developed by Apache Software Foundation is decided to use because it includes XML based marshalling and unmarshaling operations on java objects. JAXB is used to unmarshall the given XML documents to the relational Java plain java objects with respect to the given XSD schema. The most significant part of this task is to combine different XSD schemas and make JAXB work.

Schema merging phase is problematic since NVD vulnerability feeds has more than one distributed XSD schemas rather than having one concrete schema. At this point the job which is done by the automatic data collector is not sufficient to JAXB to work. These different schemas should be combined in this module. Tracing the namespaces is the tricky point for obtaining missing XSD files. The figure-14 in the below is the XSD file for the general NVD vulnerability data. It does not include the actual breakdowns within the data, however it gives reference to the other XSD files which should be traced and combined.

```

<?xml version="1.0"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns="http://scap.nist.gov/schema/feed/vulnerability/2.0"
  xmlns:vuln="http://scap.nist.gov/schema/vulnerability/0.4"
  targetNamespace="http://scap.nist.gov/schema/feed/vulnerability/2.0"
  elementFormDefault="qualified" attributeFormDefault="unqualified"
  version="2.0">
  <xsd:import namespace="http://scap.nist.gov/schema/vulnerability_0.4" schemaLocation="http://nvd.nist.gov/schema/vulnerability_0.4.xsd"/>
  <xsd:annotation>
    <xsd:documentation>TODO: address distributed with for APP->OS resolution</xsd:documentation>
    <xsd:documentation>This schema defines the structure of the National...</xsd:documentation>
  </xsd:annotation>
  <xsd:element name="nvd">
    <xsd:annotation>
      <xsd:documentation>The root element of the NVD CVE feed. Multiple "entry" child elements describe specific NVD CVE entries.</xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="entry" minOccurs="0" maxOccurs="unbounded">
          <xsd:annotation>
            <xsd:documentation>A CVE entry.</xsd:documentation>
          </xsd:annotation>
        </xsd:element>
      </xsd:sequence>
      <xsd:attribute name="nvd_xml_version" type="xsd:decimal" use="required">
        <xsd:annotation>
          <xsd:documentation>The schema version number supported by the feed.</xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
      <xsd:attribute name="pub_date" type="xsd:dateTime" use="required">
        <xsd:annotation>
          <xsd:documentation>The date the feed was generated.</xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="entry" type="vuln:vulnerabilityType">
    <xsd:annotation>
      <xsd:documentation>A CVE entry.</xsd:documentation>
    </xsd:annotation>
  </xsd:element>
</xsd:schema>

```

Figure 14 - XSD File for NVD Vulnerability Data

Once the schemas are combined, another technical problem is emerged which is the namespace collision of the inner items within the schemas. To illustrate this problem a concrete example for NVD schemas is pointed as follows; there are two schemas which are "http://nvd.nist.gov/schema/scap-core_0.1.xsd" and "http://nvd.nist.gov/schema/cpe-language_2.1.xsd". The first one is the schema of SCAP-CORE and the second one belongs to the CPE language. Both schemas have a tag named "*TextType*" within their content. The unmarshaller of JAXB cannot solve this ambiguity and gives an error for the case. At this point a manual intervention is needed and two same namespace should be resolved by giving different identifier tags to them. To perform this, the following binding.xml indicated in the figure-15 should be executed with a JAXB specified command.

```

<jxb:bindings
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:jxb="http://java.sun.com/xml/ns/jaxb"
    version="2.1">

    <jxb:bindings schemaLocation="http://nvd.nist.gov/schema/scap-core_0.1.xsd">
        <jxb:bindings node="//xs:complexType[@name='textType']">
            <jxb:class name="TextTypeFromScapCore"/>
        </jxb:bindings>
    </jxb:bindings>
    <jxb:bindings schemaLocation="http://nvd.nist.gov/schema/cpe-language_2.1.xsd">
        <jxb:bindings node="//xs:complexType[@name='TextType']">
            <jxb:class name="TextTypeFromCpe-Language"/>
        </jxb:bindings>
    </jxb:bindings>
</jxb:bindings>

```

Figure 15 - JAXB Binding Configuration

Thanks to this custom binding, complex types with the same namespaces in two different XSD schema reference can be renamed and ambiguity can be solved by this way and JAXB unmarshalling process can be executed without any error.

After the resolution of these problems, Java objects are created automatically from provided XML data feeds by data collector module. The resulting package organization, package content and class examples are given in the figures 16, 17, 18, and 19 below;

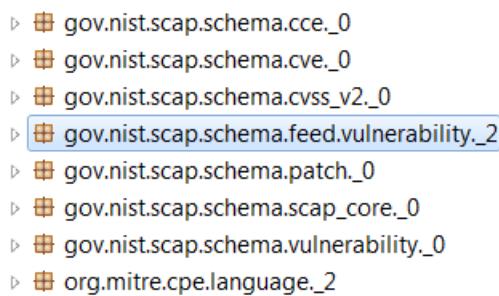


Figure 16 - Generated Relational Objects Java Package Structure

```
▲ gov.nist.scap.schema.cve_0
  ▷ CveStatus.java
  ▷ CveType.java
  ▷ ObjectFactory.java
  ▷ package-info.java
▲ gov.nist.scap.schema.cvss_v2_0
  ▷ AccessComplexityEnumType.java
  ▷ AccessComplexityType.java
  ▷ AccessVectorEnumType.java
  ▷ AccessVectorType.java
  ▷ AuthenticationEnumType.java
  ▷ AuthenticationType.java
  ▷ BaseMetricsType.java
  ▷ CiaEnumType.java
  ▷ CiaRequirementEnumType.java
  ▷ CiaRequirementType.java
  ▷ CiaType.java
  ▷ CollateralDamagePotentialEnumType.java
  ▷ CollateralDamagePotentialType.java
  ▷ ConfidenceEnumType.java
  ▷ ConfidenceType.java
  ▷ CvssImpactType.java
  ▷ CvssType.java
  ▷ EnvironmentalMetricsType.java
  ▷ ExploitabilityEnumType.java
  ▷ ExploitabilityType.java
  ▷ MetricsType.java
  ▷ ObjectFactory.java
  ▷ package-info.java
  ▷ RemediationLevelEnumType.java
  ▷ RemediationLevelType.java
  ▷ TargetDistributionEnumType.java
  ▷ TargetDistributionType.java
  ▷ TemporalMetricsType.java
▲ gov.nist.scap.schema.feed.vulnerability_2
  ▷ Nvd.java
  ▷ ObjectFactory.java
```

Figure 17 - Generated Vulnerability Java Classes

```

@XmlElementAccessorType(XmlAccessType.FIELD)
@XmlType(name = "", propOrder = {
    "entry"
})
@XmlRootElement(name = "nvd")
public class Nvd {

    protected List<VulnerabilityType> entry;
    @XmlAttribute(name = "nvd_xml_version", required = true)
    protected BigDecimal nvdXmlVersion;
    @XmlAttribute(name = "pub_date", required = true)
    @XmlSchemaType(name = "dateTime")
    protected XMLGregorianCalendar pubDate;

    /**
     * A CVE entry.Gets the value of the entry property.
     *
     * <p>
     * This accessor method returns a reference to the live list,
     * not a snapshot. Therefore any modification you make to the
     * returned list will be present inside the JAXB object.
     * This is why there is not a <CODE>set</CODE> method for the entry property.
     */
}

```

Figure 18 - Generated NVD Java Object Class

```

public class VulnerabilityType {

    @XmlElement(name = "osvdb-ext")
    protected OsvdbExtensionType osvdbExt;
    @XmlElement(name = "vulnerable-configuration")
    protected List<PlatformType> vulnerableConfiguration;
    @XmlElement(name = "vulnerable-software-list")
    protected VulnerableSoftwareType vulnerableSoftwareList;
    @XmlElement(name = "cve-id")
    @XmlJavaTypeAdapter(CollapsedStringAdapter.class)
    protected String cveId;
    @XmlElement(name = "cce-id")
    @XmlJavaTypeAdapter(CollapsedStringAdapter.class)
    protected String cceId;
    @XmlElement(name = "discovered-datetime")
    @XmlSchemaType(name = "dateTime")
    protected XMLGregorianCalendar discoveredDatetime;
    @XmlElement(name = "disclosure-datetime")
    @XmlSchemaType(name = "dateTime")
    protected XMLGregorianCalendar disclosureDatetime;
    @XmlElement(name = "exploit-publish-datetime")
    @XmlSchemaType(name = "dateTime")
    protected XMLGregorianCalendar exploitPublishDatetime;
    @XmlElement(name = "published-datetime")
    @XmlSchemaType(name = "dateTime")
    protected XMLGregorianCalendar publishedDatetime;
    @XmlElement(name = "last-modified-datetime")
    @XmlSchemaType(name = "dateTime")
    protected XMLGregorianCalendar lastModifiedDatetime;
    protected CvssImpactType cvss;
    @XmlElement(name = "security-protection")
    protected SecurityProtectionType securityProtection;
    @XmlElement(name = "assessment_check")
}

```

Figure 19 - Generated Java Object Class for Vulnerability

As it can be seen from the package and class contents, objects are created including their class relations and primitive types attributes which are properly set by JAXB unmarshaller. As a consequence of this process whole XML data for NVD vulnerabilities can be taken with the following code snippet and send to another modules.

```

try {
    jaxbContext = JAXBContext.newInstance(ObjectFactory.class);
    Unmarshaller unmarshaller = jaxbContext.createUnmarshaller();
    nvd = (Nvd)unmarshaller.unmarshal(new File("nvdcve-2.0-2014.xml"));
} catch (JAXBException e) {
    e.printStackTrace();
}

```

Figure 20 - JAXB XML Unmarshalling Code Snippet

3.5 Automatic Triplet Generator from Relational Objects

In this work no static model is enforced for the ontology frame, in contrast the model is generated automatically which is one of the key features of the system. Dynamic modeling of ontology is a complex problem to solve and also producing successful results from an automated ontology is highly dependent on the modeling and enhancement processes. There is a need for a comprehensive corpus which covers all the terminology in security domain. In addition to this, relations should be defined in a consistent manner by preventing data vagueness. One of the most significant artifacts is the selected data input for this task. The quality of this module's results is highly dependent on the input data. The input data produced in this work is refined and enhanced in previous two modules which are automatic data collector and relational object mapping.

The following diagram indicates the inner processes within this module. The initial point for the whole process is the NVD objects that are taken from previous module. The data flows through three main processes which are functioning sequentially and produces the resulting RDF data and sends to the knowledge accumulator.

NVD object tree has a container object which is similar to the root element in an XML file. This object is represented as NVD, and it includes list of vulnerability entities which also have their basic attributes and list of reference to other objects. At this point, it useful to remind the modularity principle is explained in the system architecture of the system. In order to stick to this rule, this module should be independent of the concrete type of inputs which requires a generic design. Although, this increases the complexity of the implementation, it brings a unique feature to the system which is the reusability of this module with any type of data. This point is one of the key supporters for dynamic ontology modeling task. The figure-21 below includes the processes within this module

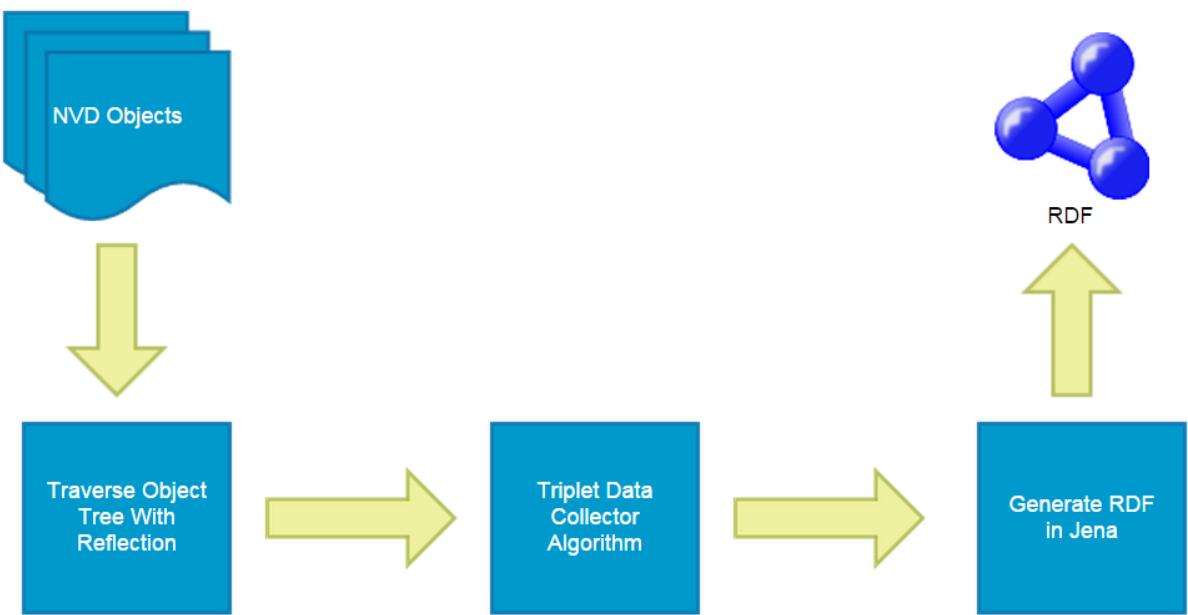


Figure 21 - Automatic Triplet Generator From Relational Objects Block Diagram

3.5.1 Triple Generation Algorithm from Relational Objects

In order to provide a type independent generic implementation, object tree should be traversed and each element should be treated as an entity with attributes and references to other entities. To overcome with this problem, reflection approach is considered as an appropriate solution. In computer science, reflection is the ability of a computer program to examine and modify structure and behavior the values, meta-data, properties and functions of the program at runtime. [27] Java reflection API is used for the implementation of the two phases "Traverse Object Tree with Reflection" and "Triplet Collector". The aim is to traverse the whole object hierarchy and extract triplets with respect to the proposed algorithm. The idea behind this algorithm constructs a bridge between object-oriented data structure and RDF triple structure. As explained in the background section, an RDF triple is composed of three nodes which are the "subject", "predicate" and "object". To illustrate the similarity between the structures of OO and RDF the following example can be examined which is taken from NVD data set. CVE-2014-3296 is a vulnerability which is represented as *VulnerabilityType* object. This object has a reference to *CvssImpactType* object which does not have any property, however this object extends from another object which is *CvssType* and this type contains list of *BaseMetricsType*. One this *BaseMetricsType*

object includes an *AuthenticationType* which has a value of *AuthenticationEnumType* which has the value *SINGLE_INSTANCE*. The hierarchy is illustrated in the following class diagram with UML (Unified Modeling Language) notation in figure 22.

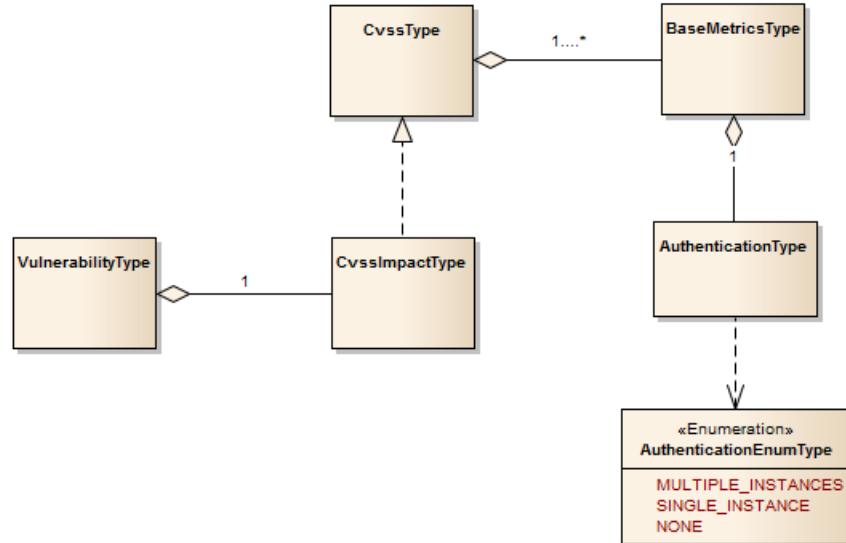


Figure 22 - Class Diagram for Vulnerability Type

In the class diagram the relations between classes are indicated as aggregation, realization and dependency. The corresponding RDF triple structure for this model is represented in the following figure-23;

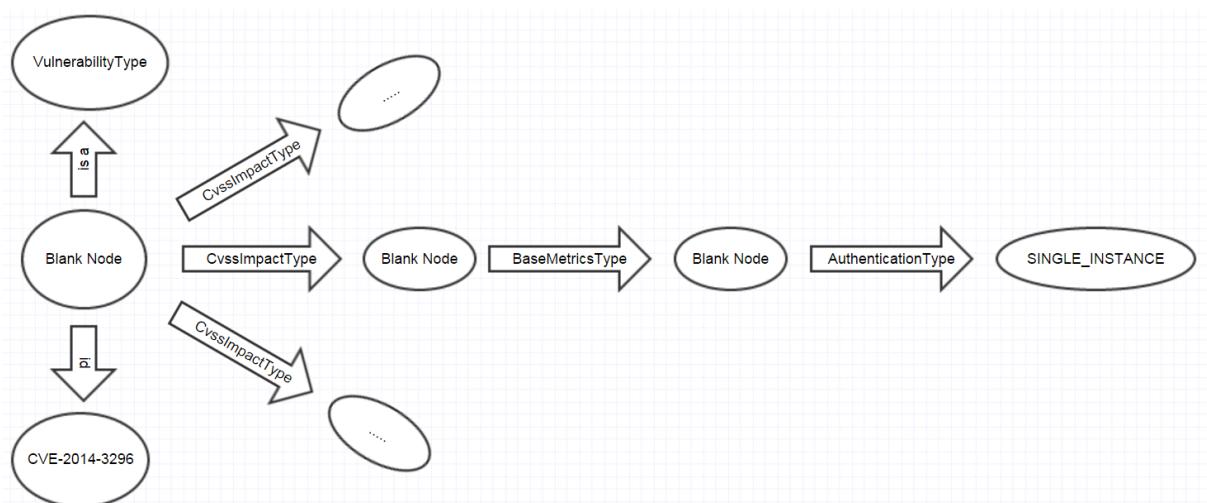


Figure 23 - RDF Model for Vulnerability Type

Blank nodes are necessary for representation of the in RDF format, since each relation does not necessarily have a value in object tree. Some relations represent the references for relations and those are indicated as blank nodes. In this representation ellipses are the concepts and arrows are the predicates. The direction of these arrows is not significant, therefore this linked data representation can be considered as an undirected graph which can be traversed without any sense of direction. As a result with a simple SPARQL the following questions can be answered; "*What is the id of vulnerability types whose authentication type is single instance*" or "*Which vulnerability types have base metrics including single instance authentication*". The possibility of the queries increases as the graph is populated with more concepts and relations. At this point the important point is the ability to answer to dynamically changing queries which means no predefined query should be determined, since no static model exists, the linked data has the ability of dynamically adapt the new data and accumulates it in order to answer modified queries in time.

To perform a conversion from relational object to RDF triples, an algorithm is developed under the scope of this work. The following figure-24 represents the pseudo code of the algorithm which is originally developed in java language. Sub methods are omitted for the brevity of the main algorithm. Those are given in the appendix-A.

```

generateRDF (Object)

    IF object is not a JAXB entity THEN
        RETURNS null

    classType = CALL object.getClass RETURNS Class<?>
    idValue = null

    method = CALL classType.getMethod WITH "getId" RETURNS Method
    idValue = CALL method.Invoke WITH object RETURNS String
    resource = CALL createInheritRDFResource WITH (CALL object.getClass.getSimpleName)

    IF idValue is not null THEN
        fieldValue = CALL idValue.toString RETURNS String

    fields = CALL getAllFields WITH object

    FOR each field in fields
        propertyName = field.name

        IF field.classType is type of List class THEN
            IF field.obj is null OR field.obj is not type of List<Object> THEN
                CONTINUE

            listObj = field.obj
            iterator = Call listObj.iterator RETURN Iterator<Object>

            WHILE CALL i.hasNext RETURNS boolean
                element = CALL i.next RETURNS Object

                IF element is JAXB entity THEN
                    primObject = element
                    p = CALL createRDFFProperty WITH propertyName RETURN Object
                ELSE
                    res = CALL generateRDF WITH element
                    IF res is not null THEN
                        CALL addResourceToRDFResource WITH resource, propertyName, res
                ELSE
                    IF field.obj is not JAXB entity THEN
                        primObj = field.obj
                        p = createRDFFProperty WITH propertyName RETURNS Property
                        CALL resource.addProperty WITH p, primObj.toString
                    ELSE
                        res = CALL generateRDF WITH field.obj RETURNS Object
                        IF res is not null THEN
                            CALL addResourceToRDFResource WITH resource, propertyName, res
                ENDIF
            ENDWHILE
        ENDIF
    ENDFOR
ENDIF

RETURN resource

```

Figure 24 - RDF Triple Generator Algorithm From Relational Objects

This algorithm is called for root object. The steps performed within the algorithm are summarized below;

- Take the root object as an input to generation algorithm
- Starting from the parent recursively perform the following operation for each object
 - Create id triplet
 - Create type triplet
 - Get fields of the object
 - For each field check the field type, iterate for each element and perform the earlier steps if the field is type of list
 - Create triplet if the field is type of an object
 - Recursively perform the steps in upwards for the object

Resource and Property types are taken from JENA libraries for ontology generation. The RDF creation logic is hidden behind `AddResource` and `AddProperty` method implementations which does not included above. Once the algorithm is started, it recursively created RDF triples and add to the JENA ontology model instance which is send to the final module called Knowledge Accumulator.

3.6 NLP Oriented Information Extractor

The concept of making raw data useful has been a significant research topic in computer science related fields. *Information Extraction* is the key task to achieve this issue. This task is based on extracting structured information from both semi-structured (machine readable documents) or unstructured data. Conversion and mapping methodologies can be used for the semi-structured data. However in case of the unstructured inputs, an extra problem is faced since machines do not recognize unstructured data which is the human language. In order to make language understandable by automated systems, syntactical analysis should be performed which are regarded as NLP techniques.

One of the major problems defined in the scope of this work is to enhance the proposed ontology by utilizing from domain related unstructured data which are mainly the summary and description fields of the NVD items. Summary fields of each CVE entry include

sensational data about the vulnerability like “Who does it?” “How does it occur?” “Which is critical information in operational degree?” The NLP Oriented Information Extractor module is developed for the extraction of this sensitive data and contributes to the enhancement of the knowledge base.

In computer science, NLP approach includes numerous major tasks which should not be necessarily part of the solution; however each task contributes to the overall success when appropriately used. In the scope of the system the information extraction approach is divided into two major tasks which are the information extraction from the given unstructured text and the scoring of the extracted phrases. The aim of the first task is to extract phrases with their relations which can be regarded as the candidate phrases. Scoring task completes the extraction process by eliminating the irrelevant candidates and enhances the whole process.

3.6.1 Information Extraction from Unstructured Inputs

As it is stated in the earlier, the data collector and differentiator module separates the related unstructured texts and send them to this module for information extraction. The majority of the input is composed of the summary texts from NVD vulnerabilities which are also used here to explain the whole internal information extraction processes. The development phase of this module is initiated with the examination of NVD vulnerability summaries. The work done in [\[15\]](#) is also utilized from CVE vulnerability summaries by expressing summary text structure with a simple Extended BNF (EBNF). This expression parses the given vulnerability and extracts concepts from such predefined relation rules; "because of", "when", "in" and "allow". This approach seems very effective for particular summary fields of vulnerabilities, however this structural analysis does not guarantee for responding further modifications within summary structures. Therefore such an expression is not sufficient for the generic use and not preferred within the scope of this module.

The proposed methodology in this module is *Semantic Parsing* which is used to process the given grammar syntactically and tags part of speech. This approach includes phrase chunking and part-of-speech methodologies which are applied sequentially to process the given raw text. After the text is parsed and phrases are tagged, the proposed extractor algorithm defines concepts with their relation. There is an extra step which is called *Phrase*

Ranking applied to the tagged phrases, especially the noun phrases which represent a resource within the ontology. The lists of processes are displayed in the following figure-25 for the *Information Extraction* phase;

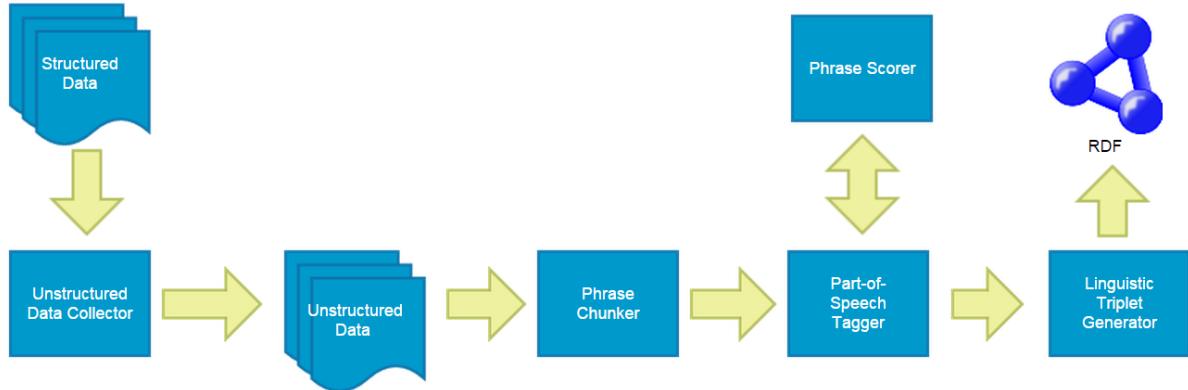


Figure 25 - Information Extraction Module Block Diagram

Part-of-speech tagging is a process where tokens are sequentially labeled with syntactic labels such as noun, verb, adjective, etc. There exist different types of tagging methods which are Rule-Based POS Tagging, Transformation-Based Tagging, and Stochastic (Probabilistic Tagging). These tagging techniques are applied to the selected word or phrase.

At this point the difference between a single token and composite token is significant for tagging process. Therefore n-gram technique is applied to perform the possibility of token combinations. An n-gram is a subsequence of n items from a given sequence. The following example illustrates the n-gram methodology [28]:

Example sentence is "... to be or not to be ..." and corresponding n-grams are indicated as follows;

Unigram => to, be , or, not, to, be

Bigram => to be, be or, or not, not to, to be

Trigram => to be or, be or not, or not to, not to be

Combinations of tokens are proportional to the number of "n" in n-grams technique. In this work this is one of the variables that used interchangeably for different cases in order to improve semantic parsing success.

Unstructured text inputs are analyzed by using Apache OpenNLP which is a java based technology developed by apache community. It is a good modern and robust alternative for Lingpipe which is quite old and has limited feature set for the implementation phase. In the semantic parsing operations both Phrase Chunking and Part-of-speech tagging are used in hybrid manner. OpenNLP parser firstly parses the data by separating it to tokens and labels the tokens with respect to their factor within the given sentence. Tokenization method is based on n-grams methodology and the Part-of-speech algorithm is based on the OpenNLP corpora (trained model). No other predefined corpus is used for detection.

To illustrate the semantic parsing operation, an example is taken from the OpenNLP documentation within Apache Open NLP web site. The classic text which will be processed by parser is as follows;

The quick brown fox jumps over the lazy dog .

The parser processes this input text and produces the following tokens with POS tag values;

(TOP (NP (NP (DT The) (JJ quick) (JJ brown) (NN fox) (NNS jumps))
(PP (IN over) (NP (DT the)
 (JJ lazy) (NN dog))) (. .)))

As it is indicated above, there are cascading parentheses including the part-of-speech tag content. To solve the meaning of the abbreviations of POS tags Part-of-Speech Tagging Guidelines for the Penn Treebank Project [\[29\]](#) is used. The following table represents some of these tags and corresponding abbreviations;

NP	Noun Phrase
DT	Determiner
JJ	Adjective
NN	Noun, singular or mass
VBZ	Verb, 3rd person singular present
PP	Pronoun

Table 2 - Part of Speech Tag Definitions

An algorithm is developed for the linguistic triplet generation process in conjunction with the above techniques provided by OpenNLP framework. The aim of the algorithm is to extract concepts with their relations from the given text. Noun extraction is not sufficient for this processes, adjectives, propositions and verbs play a significant role to bound these extracted phrases and convert them to the RDF triple format which consists of subject-predicate-object structure. The idea behind the algorithm is based on the synonym between linguistic patterns and RDF structures. Verbs are the first candidate for the predicate role, however prepositions should be considered in a different context. Prepositions may refer to various points in a sentence, for instance when the following noun phrase "*REST API in oVirt 3.4.0*" is examined, the preposition "*in*" is used to indicate that the "*REST API*" resides in "*oVirt 3.4.0*" which should be represented as a whole concept (subject or object) in RDF. However for the following noun phrase "*to obtain sensitive information via a crafted web page*", the preposition "*via*" is used to denote the approach used for obtaining sensitive information, so in this case division of the phrases "*sensitive information*" and "*crafted web page*" makes more sense. Developed algorithm relies on the second example, and performs a division in the first example as well. Since the content of the generated concepts should be defined generic as much as possible. This approach is seemed to urge the ontology from lacking of precision by losing more specific phrases and representing them with a concept group, however this argument is invalid since representing a concrete phrase through more than one generic phrase ease the burden of further semantic reasoning and analysis processes. The nodes in the ontology graph should be decoupled a represent the whole hierarchy.

In the light of these considerations, the algorithm for this process is defined as follows;

- First tokenize sentence and label each word combination with corresponding element role (verb, phrase, noun ..)
- Set a threshold for eliminating long phrases (threshold is varies between 4 -10 words)
- Set relative position of each phrase within sentence
- Take the following patterns with respect to the given pseudo-code;

```

current = firstPhrase;

WHILE current is not null
  IF current.next OR current.next.next == null THEN
    |
    | BREAK;
    |
    | IF ((CALL IsPrePosition WITH current.next.type OR CALL ISVerb WITH current.next.type) AND CALL IsNoun WITH current.next.next.type) THEN
    |   CALL GenerateTriplet WITH current.text, current.next.text, current.next.next.text
    |
    | current = current.next

```

Figure 26 - Semantic Parser Algorithm Pseudo Code Snippet

The following sentence is taken from NVD vulnerability summaries. The text is given as input to this process and the results are indicated accordingly. “S” denotes for the subject form, “P” denotes for the predicate and “O” indicates the object form in the RDF representation.

Vulnerability summary text;

"The REST API in oVirt 3.4.0 and earlier stores session IDs in HTML5 local storage, which allows remote attackers to obtain sensitive information via a crafted web page"

Extracted phrases are indicated as follows;

REST API(S) in(P) oVirt 3.4.0(O)

The REST API in oVirt 3.4.0 and earlier stores session IDs in HTML5 local storage(S)

*The REST API in oVirt 3.4.0 and earlier stores session IDs in HTML5 local storage(S) allows(P)
remote attackers(O)*

remote attackers to(S) obtain(P) sensitive information(O)

sensitive information(S) via(P) a crafted web page(O)

Corresponding RDF representation of the unstructured text is indicated in the figure-27;

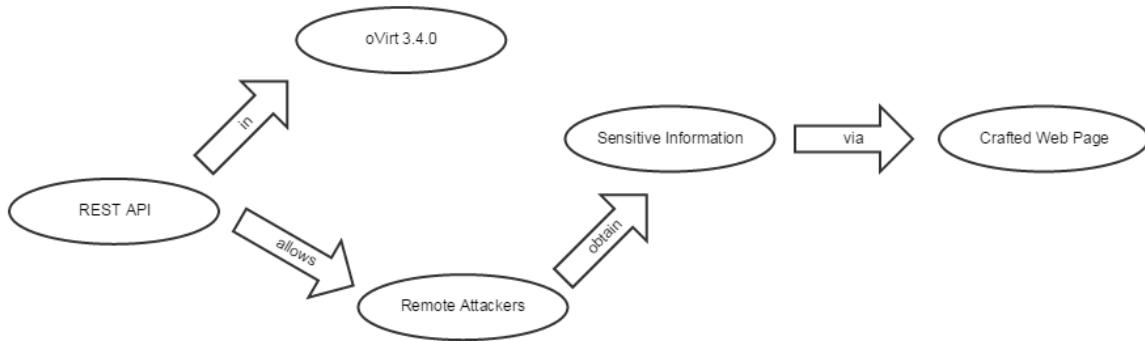


Figure 27 - RDF Representation of Parsed Vulnerability Summary

3.6.2 Phrase Scoring

The success of the extraction phase is required to support with an external ranking mechanism. To process only the description part of vulnerability produces numerous candidate phrases. It is an undeniable fact that extracting concepts without an appropriate filtering mechanism increases the total coverage of the ontology; however this is not the desired aim from the perspective of this system. Cyber security is the main domain of the system in which vulnerabilities, threats, assets, countermeasures and related stuff should be placed. To link to these items there should be additional concepts and relations which consolidates the generated ontology. However the line between relevancy and vagueness should be discriminated professionally which means a domain expert of cyber security should not be overloaded with the unnecessary data. This situation decreases the precision of the reasoning processes based on the generated ontology and fails the system. An intelligent elimination of the candidate concepts should be performed without rushing into extremes. The proposed phrase scoring block diagram is indicated in the following figure-28;

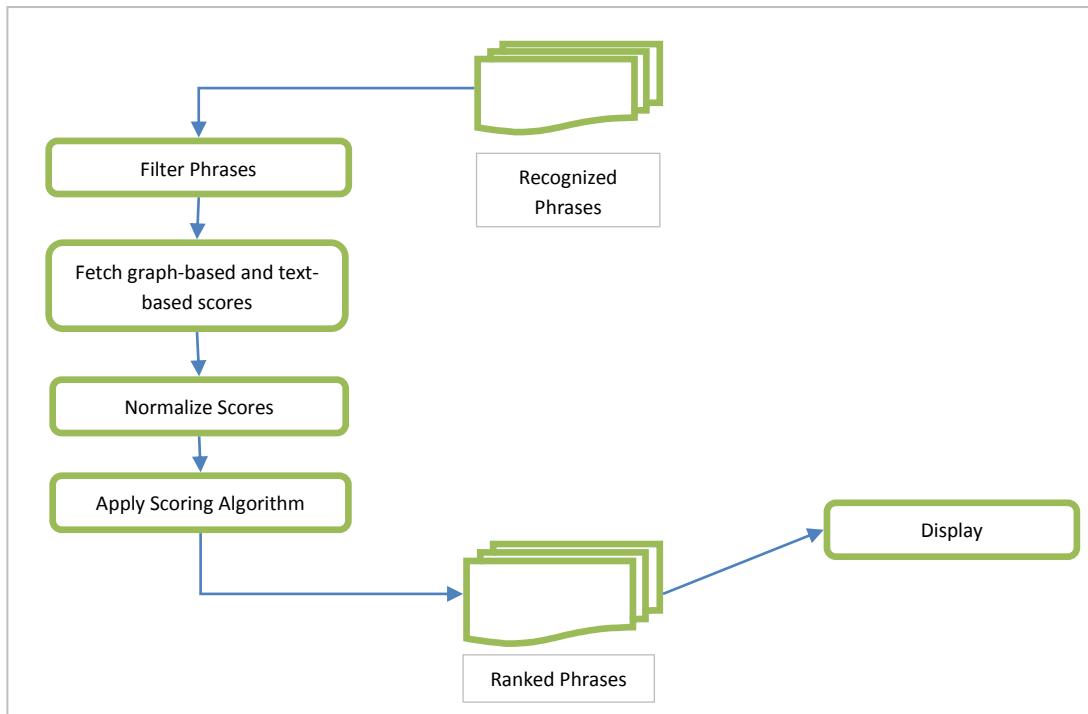


Figure 28 - Phrase Scoring Block Diagram

Phrase ranking process includes computing a score for each concept extracted from text and ranking the concepts according to their scores. Scoring methods can be divided into 2 different groups, text based scores and graph based scores. Text based score are computed by statistical information that is generated from text. Graph based scores are calculated with the help of relations defined between concepts.

Text based measurements are also examined in two different groups within the scope, first one is the *NLP based Statistical Measurements* including similarity calculations which play an important role in key phrase extraction among many neighboring items. The second division is the *Document Index based* measurements which are interfered with the term document occurrences within the selected corpora. This corpus can be selected as trained or non-trained set document indexes. Both of these methods are explained in the under the following Text based statistical measurements section.

3.6.2.1 Text Based Statistical Measurements

Term/Document statistical measurements are including both term level and document level statistical calculations within a given text or the given corpus. In the figure below,

statistical measurements used for scoring are stated based on [21]. Additionally, Word Frequency and Head Noun Frequency measurements are stated here with their descriptions.

Name of Feature	Description
Stemmed phrase	the stemmed form of a phrase, for matching with human generated phrases
Whole phrase	the most frequent whole (unstemmed) phrase corresponding to stemmed_phrase,
Number of words in phrase	the number of words in stemmed phrase, ranging from one to max
First occurrence of phrase	the first occurrence of stemmed phrase, normalized by dividing by the number of words in the document (including stop words)
First occurrence of word	the first occurrence of the earliest occurring single stemmed word in stemmed phrase, normalized by dividing by the number of words in the document (including stop words)
Phrase frequency	the frequency of stemmed phrase, normalized by dividing by the number of words in the document (including stop words)
Word frequency	the frequency of the most frequent single stemmed word in stemmed phrase, normalized by dividing by the number of words in the document (including stop words)
Head Noun Frequency	The frequency of the head noun of the phrase in the document.
Relative length	the relative length of whole phrase, calculated as the number of characters in whole phrase, divided by the average number of characters in all candidate phrases
Proper noun	is whole phrase a proper noun, based on the capitalization of whole phrase?
Final adjective	does whole phrase end in a final adjective, based on the suffix of whole phrase?
Common verb	does whole phrase contain a common verb, based on a list of common verbs?
TF-IDF	Term frequency – inverse document frequency

Dispersion	<p>the clumping property named as “dispersion” of a phrase is calculated as in the following:</p> $E = D \times \left[1 - \left(1 - \frac{1}{D} \right)^T \right]$ <p>where D is the number of documents in the corpus, and T is the total number of occurrences of the phrase.</p> <p>The dispersion would be one if the phrase is randomly distributed over documents. The dispersion would be less than one if it is clustered.</p>
------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Table 3 - Text Based Statistical Measurements

The statistical measurements described above are classified under three categories in this module. There is weighted scoring process which gives parametric coefficient values for each of the measurement type. This provides a stable evaluation bench which can be executed with different parameters and comparison of the results by modifying the corresponding measurement weights. Each score is normalized after the calculation with respect to a bias value in order not to destroy effectiveness of some measurements.

First category is the relative text measurements which does not need any external data to calculate. The following measurement belongs to this group; Number of words in phrase, First occurrence of phrase, First occurrence of word, Phrase frequency, Word frequency, Head Noun Frequency and Relative length. This weight of this category to the general score of the candidate phrase is less than the other categories since the importance degree of a phrase is not directly depends on the relevancy to the given text in case of ontology development. Second category consists of linguistic measurements which are also provided by OpenNLP framework as explained in the NLP Oriented Data Extractor module. Those are Stemmed phrase, Whole phrase, Head Noun Frequency, Proper noun, Final adjective and common verb. The weight of these calculations is also not very effective in final scoring since they share the same reason for the first category.

Third category is the smallest group which is composed of TF-IDF and Dispersion. These measurements are different in the sense of calculation phase. Usually and external data

source should be required which acts as a corpus. TF-IDF is a numerical statistic that indicates the degree of importance of a particular word is to a document in a corpus. The formula seems very basic but it is used in numerous of information retrieval based systems. The formula is defined as follows; t denotes for a term within a document inside of the corpus, d denotes the document which includes term t and D denotes the all documents within the corpus. The formula is the multiplication of text frequency and inverse document frequency [\[30\]](#).

$$tf(t, d) = 0.5 + \frac{0.5 \times f(t, d)}{\max\{f(w, d) : w \in d\}}$$

[Equation 1 - Text Frequency Calculation Formula](#)

$$tfidf(t, d, D) = tf(t, d) \times idf(t, D)$$

[Equation 2 - Term Frequency–Inverse Document Frequency Calculation Formula](#)

In the above formula $tf(t,d)$ is the frequency of the text within the given document. High frequency is a positive sign for the relevancy of the word within the given document. However it is meaningless without the $idf(t,D)$ calculation since this part of the formula is the main decider about the relevancy degree of the given word. Inverse document frequency is calculated with different ways, the common one is as follows in equation-3 [\[30\]](#);

$$idf(t, D) = \log \frac{N}{|\{d \in D : t \in d\}|}$$

[Equation 3 - Inverse Document Frequency Calculation Formula](#)

In the above formula N denotes the total number of documents in the corpus, d is the document which term t is contained by and D denotes the whole documents within the corpus. It can be clearly inferred that the idf value is proportional to the minority of the documents which contains the given term within the corpus. Therefore, stop words which are common in all documents has very little tf-idf value compared to the unique words like domain terms. Dispersion is the other measurement which indicates the homogeneity among the general distribution of the given term within the given corpus. This value can be used to indicate whether the word is used randomly or clustered within the documents which indicate its significance.

In order to use the measurements defined in the third category, an additional subsystem is needed to utilize from external data sources. At this point Wikipedia is decided to use as an external corpora. The following section identifies how the content and metadata of Wikipedia sources are used.

3.6.2.1.1 Wikipedia Data Source Utilization

World's common knowledge repository Wikipedia is currently the largest free web encyclopedia. The comprehensiveness of the Wikipedia comes from its large number of active contributors which exceeds 91.000. Wikipedia accommodates over 3.5 million articles in English. [31] The development of the system is conducted by volunteers around the globe collaboratively. Article contents can be edited by users and checked by system administrators in time to prove its validity and correctness. The system dynamics depends mostly on users' contributions, therefore almost any kind of contemporary information can be found as well as scientific articles in the scope of Wikipedia. There is also a multi-language support in the system and most of the articles are also present in other languages. Although, articles can easily be changed by any users, there is a strict editorial control on changed contents in terms of style, verifiability, reliability and citations.

Wikipedia is the selected reference source for text based measurements. It is a big corpus containing both relevant and irrelevant data about cyber security. However it is a good reference point for the general text processing method with the contribution of its coverage for the terms. Wiki Dumps have very comprehensive data for each wiki page

including title, author, reference and content information. The concepts used in NVD or the texts taken by its feeds shares a common information at some point within the Wikipedia dump data which urges this work to perform a matching the extracted concepts from cyber security data to the Wikipedia page titles or entities covered by these pages in certain fields. This approach provides to identify information about the wiki description of the concepts, category of the concepts, and number of occurrence within the corpus or within the specific concepts descriptions. These capabilities are very beneficial for the ranking algorithm of this module.

The principle requirement of this approach is to compute the frequency of co-presence concepts. This yields an indicator value showing how often those concepts are occurring together or individually in entire corpus. The measure directs this study to make inferences about concepts similarity degree. Wikipedia's broad coverage of information provides a homogenous inverted index which gives clue about any kind of concept, however it might also incorporates unnecessary ones which affect the occurrence frequencies any may be misleading in index frequency calculations.

Wikipedia dumps are publicly released without any required license for individual users. The main problem that should be addressed here is to develop a methodology for an efficient use of this data. A typical wiki dump is about 28 GB in xml format with page revisions which are ignored in the scope of this work. In order to manage with this data efficiently and in a professional manner, it is decided to use an indexing engine which is *Apache SolR*.

The processes done while indexing Wikipedia and querying this index is shown in the following figure-29;

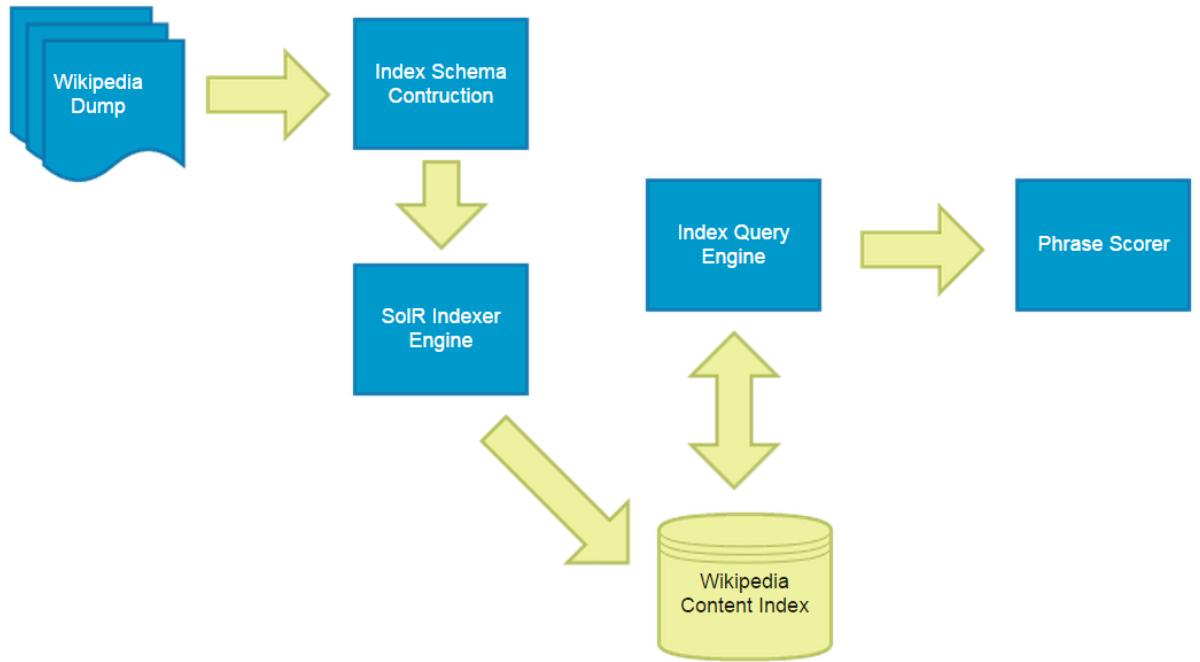


Figure 29 - Wikipedia Index Utilization Block Diagram

The process initiated with the collection of the selection Wikipedia dumps. Then the installation of SolR engine is performed which is explained in the web page tutorial in detail. There should be an independent SolR server engine should be run the machine. After the server side awakens, the admin panel can be browsed from local host. The next phase is to construct the index schema for Wikipedia data dump model and configure the corresponding schema files in the SolR engine files. The schema for Wikipedia dump indexing is developed and stated within the scope of this work. The following figure-30 represents the "*dataconfig*" file of the SolR engine that indexes any given Wikipedia dump without any problem. The columns in the configuration file represent the fields that exist in the dump file. These columns are mapped with the "*xpath*" tag to the wiki locations.

```

<dataConfig>
    <dataSource type="FileDataSource" encoding="UTF-8" />
    <document>
        <entity name="page"
            processor="XPathEntityProcessor"
            stream="true"
            forEach="/mediawiki/page/"
            url="../core/exampledocs/enwiki-20140903-pages-articles.xml"
            transformer="RegexTransformer,DateFormatTransformer"
        >
            <field column="id" xpath="/mediawiki/page/id" />
            <field column="title" xpath="/mediawiki/page/title" />
            <field column="revision" xpath="/mediawiki/page/revision/id" />
            <field column="user" xpath="/mediawiki/page/revision/contributor/username" />
            <field column="userId" xpath="/mediawiki/page/revision/contributor/id" />
            <field column="text" xpath="/mediawiki/page/revision/text" />
            <field column="timestamp" xpath="/mediawiki/page/revision/timestamp" dateDateTimeFormat="yyyy-MM-dd'T'hh:mm:ss'Z'" />
            <field column="$skipDoc" regex="^#REDIRECT .*" replaceWith="true" sourceColName="text"/>
        </entity>
    </document>
</dataConfig>

```

Figure 30 - Schema for Wikipedia Indexing in SolR

Indexing process takes approximately ten hours within the development machine which is a reasonable time since indexing processes is performed once for the initial phase only. Approximately nine million documents are added to the index which can be seen in the following figure from admin panel;

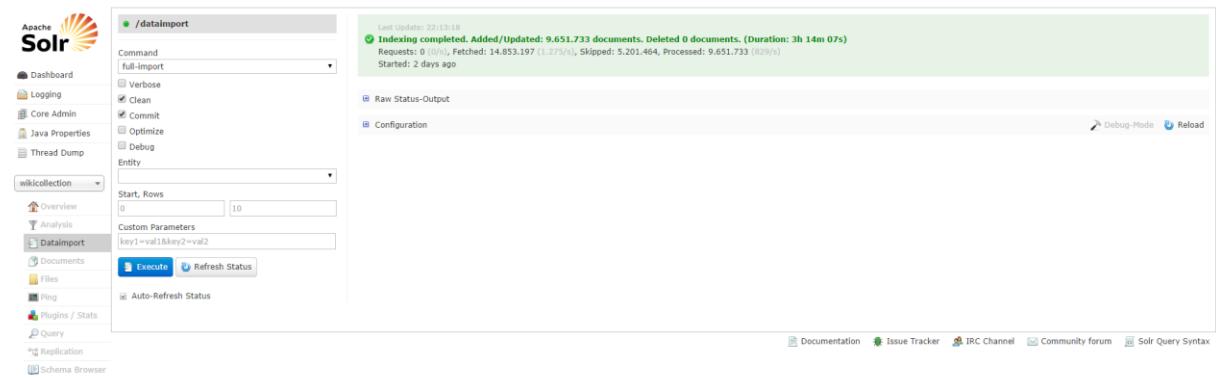


Figure 31 - SolR Admin Panel for Wikipedia Index Results

After the index is constructed successfully, querying operations can be performed in milliseconds among full index. The components which are defined in the formula of TF-IDF calculation is processed by querying the Wikipedia index on SolR. Details of the querying operations in SolR are indicated in the appendix-d.

3.6.2.2 Graph Based Measurements

Corpus utilization is a value-adding process for this study, since the underlying idea of phrase scoring mechanism is the utilization from external reference knowledge bases that contributes to the selection of relevant concepts and relations. For text based statistical measurements, Wikipedia data dumps are fully utilized by using the huge information included. However, the aim of this study is not only utilize from the external information but also the system is designated to benefit from the advantages of external knowledge bases too. To realize this idea, external knowledge bases are examined from the perspective of coverage, usage and relevancy. Among the examined knowledge bases, two major ontologies are decided to use which are DBpedia Ontology [\[32\]](#) and Freebase Ontology [\[33\]](#). Especially the enormous size of the Freebase extends the scope of this study which increases the data enrichment degree and comprehensiveness; however this also reduces the maintainability and stability of the system. In this study, Freebase data dumps are utilized and structured an ontological graph which meets the need of external knowledge base. More details are given about DBpedia and Freebase in appendix-b and appendix-c correspondingly.

3.6.2.2.1 Freebase RDF Graph Construction and Utilization

Freebase has 1.9 million triplets which reside in their public data dump published on the Freebase web site [<https://developers.google.com/freebase/data#freebase-rdf-dumps>]. The size of these reaches 22GB in “Gzip” format and 250 GB when decompressed. [\[34\]](#) To handle with this size of data, the “Gzip” formatted data is not decompressed and parsed with this compressed format in order to improve the performance. The RDF data in Freebase dump is serialized using the N-Triples format, encoded as UTF-8 text. Therefore, N-Triples format parser is developed in order to parse the data in dumps and convert it to the JENA RDF model which is commonly used for knowledge accumulation within the scope of this study.

N-Triples is a line-based, plain text format for representing the correct answers for parsing *RDF/XML[RDFMS]* [\[35\]](#). The following N-Triples are indicated as an example for this format;

```

:<http://www.w3.org/2001/sw/RDFCore/ntriples/> <http://purl.org/dc/elements/1.1/creator> "Dave Beckett" .
:<http://www.w3.org/2001/sw/RDFCore/ntriples/> <http://purl.org/dc/elements/1.1/creator> "Art Barstow" .
:<http://www.w3.org/2001/sw/RDFCore/ntriples/> <http://purl.org/dc/elements/1.1/publisher> <http://www.w3.org/> .

```

Figure 32 - N-Triples RDF Representation of Freebase Linked Data

Three RDF statements are included in the above format which represents the following RDF/XML;

```

<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
           xmlns:dc="http://purl.org/dc/elements/1.1/">
  <rdf:Description rdf:about="http://www.w3.org/2001/sw/RDFCore/ntriples/">
    <dc:creator>Art Barstow</dc:creator>
    <dc:creator>Dave Beckett</dc:creator>
    <dc:publisher rdf:resource="http://www.w3.org/" />
  </rdf:Description>
</rdf:RDF>

```

Figure 33 - XML Representation of N-Triples Format

The parser scans the freebase compressed dumps and convert them to the form of <subject>, <predicate> and <object>. There are some constraints about subject, object and predicates which should be taken into consideration in the parser utility. These constraints are stated by Freebase as follows;

The subject is the ID of a Freebase object. It can be a Freebase MID (ex. m.012rkqx) for topics and CVTs or a human-readable ID (ex. common.topic) for schema. [\[34\]](#)

The predicate is always a human-readable ID for a Freebase property or a property from a standard RDF vocabulary like RDFS. Freebase foreign key namespaces are also used as predicates to make it easier to look up keys by namespace. [\[34\]](#)

The object field may contain a Freebase MID for an object or a human-readable ID for schema from Freebase or other RDF vocabularies. It may also include literal values like strings, booleans and numeric values. [\[34\]](#)

In the light of the statements above, the parser converts the Freebase dumps into the triplet from which can be utilized by the JENA RDF model of this study. The important point is to externalize these converted triples from the existing knowledge base of the system,

since this data includes a wide range of topics from different domains which reduces the usability of the Cyber Security ontology and makes the whole effort in vain. Therefore, this triple collection should be utilized as an external knowledge base in order to provide graph based measurements. At this point JUNG graph framework is introduced which is used for construction of graph DB within the memory by directly attaching the RDF model generated by JENA.

JUNG stands for the Java Universal Network/Graph Framework which is an open source graph modeling and visualization framework developed by the Java community. Additional benefits of the Jung library are its provided graph analysis algorithms such as graph clustering and metrics for node centrality. This framework provides numerous graph utilities for graph theory, data mining, network analysis and clustering. To use these functionalities a graph structure should be constructed by using this framework or it should be imported externally. RDF representations perfectly fit to the graph representations in abstract manner. Relations can be considered as edges and concepts can be defined with the vertices. In the graph representation of an RDF structure, edges has a direction from the subject to the object, therefore the graph can be considered as a directed graph, however from the perspective of querying, the reverse directions can also be utilized within the semantic graph. Since the RDF structure is modeled in JENA the need for a corresponding JUNG representation in order to fully utilize from JUNG.

To consolidate a bridge between JENA and JUNG framework, a wrapper is introduced by shellac called *JenaJung* [36]. The source code of the project is publicly released in GitHub and can be used with the Jena core library version 2.12.1 which is used in the implementation parts of this study. By using this third party library, models defined in JENA can be converted to the JUNG graph structure.

The following graph segment illustrates the semantic graph structure constructed from external knowledge base;

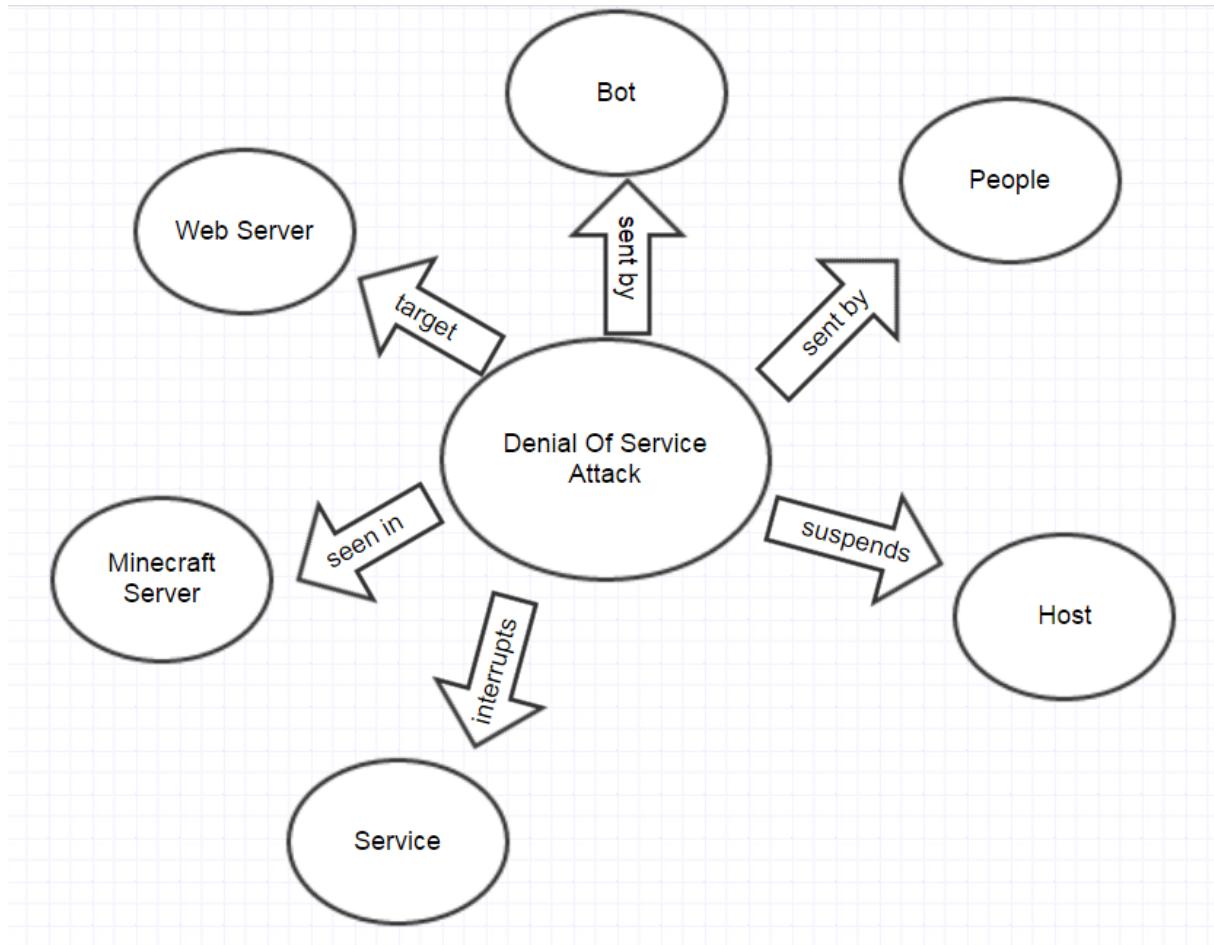


Figure 34 - Knowledge Representation of the Concept "Denial of Service Attack" in Freebase Graph

Each concept represented above corresponds to a topic within the Freebase data source with included relations. After the conversion, these concepts and relations can be directly mapped with the extracted phrases and scoring algorithms can be applied in a straightforward manner.

For the graph based scoring calculation the *PageRank* algorithm is used which is also regarded as "Google Search Algorithm". It is originally developed by Google Inc. to rank websites in their search engine results. The web sites and their links are represented in a graph structure and the algorithm is applied to each node (web-site) within the graph. "The algorithm works by counting the number and quality of the links to a page to determine a rough estimate how important it is" [37]. The tendency of the algorithm is increasing the weight of node if it is linked by many other nodes which can also be considered in a social environment. People who have more relationships from other people are regarded as popular and their importance level is increased accordingly. The rule is similarly applied by

the PageRank algorithm and Google web search is still based on this algorithm. The following figure illustrates the PageRank algorithm as follows;

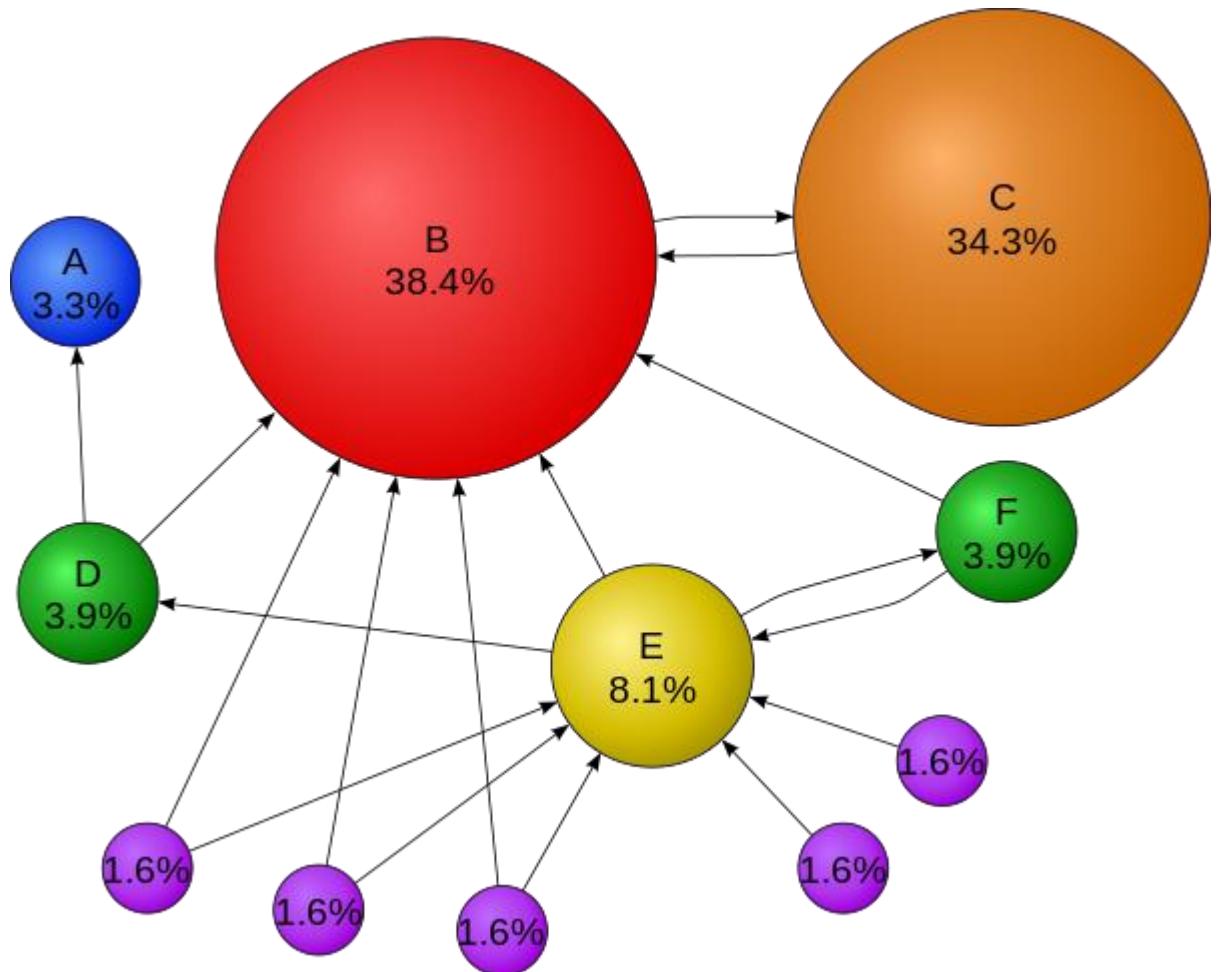


Figure 35 - Page Rank Algorithm Illustration

The algorithm is utilized by the JUNG library as it is indicated below, however the definition is also put here for information [\[38\]](#);

$$PR(A) = (1-d) + d \left(\frac{PR(T_1)}{C(T_1)} + \dots + \frac{PR(T_n)}{C(T_n)} \right)$$

where

$PR(A)$ is the PageRank of page A,

$PR(T_i)$ is the PageRank of pages T_i which link to page A,

$C(T_i)$ is the number of outbound links on page T_i and

d is a damping factor which can be set between 0 and 1.

The final formula can be represented as follows;

$$PR(p_i) = \frac{1-d}{N} + d \sum_{p_j \in M(p_i)} \frac{PR(p_j)}{L(p_j)}$$

[Equation 4 - PageRank Calculation Formula](#)

where p_1, p_2, \dots, p_n are the pages under consideration, $M(p_i)$ is the set of pages that link to p_i , $L(p_j)$ is the number of outbound links on page p_j , and N is the total number of pages.

[\[38\]](#)

Utilization of PageRank algorithm in this study is a very beneficial factor for the enhancement of the knowledge base, since it directly affects the scoring mechanism and provides a good reference point for extracted concepts and relations. For instance in the figure-34, the page rank of the concept "DOS Attack" is calculated with respect to the neighboring concepts among which are indicated as "Host", "Service", "Web Server", "Bot" etc. The more DOS attack is seen by any system or related with any product, the higher its page rank value will get. The popular concepts will be highly appreciated by the knowledge base of this study, since the input source is highly related to the domain of the Cyber-Security which does not discriminated in the context of the Freebase RDF graph. If the input source of this system is not from the cyber security domain, in other words the input collector module is deactivated and any text source will be considered as input then the page ranking scoring emphasizes numerous concepts which are not relevant with the cyber security domain which causes the vagueness of the ontology.

CHAPTER 4

SYSTEM EVALUATION

Information retrieval tasks, especially NLP oriented systems have always been difficult to evaluate with a standard reference point because of the domain dependency and human language interference. The evaluation of the system is separated into two phases with respect to the input types which are structured and unstructured sources.

4.1 Evaluation of Knowledge Base from Structured Data

The first phase of the system is very substantial which constitutes the stable part of the system. In order to verify extracted triples from structured source, an OWL file verification tool should be used. However the OWL file is not generated directly in the scope of this system, in contrast it is generated via JENA framework that does not need any owl file. The benefit of using JENA is to be exempted from any owl verification method, since it is already verified. The following table includes the relations extracted from NVD 2014 CVE data feed xml;

summary	47
deprecationDate	1415
cwe	28
lang	98027
vulnerableConfiguration	33
id	59
references	21054
title	97882
vulnerableSoftwareList	33

logicalTest	48
referenceType	126
any	92295
name	96057
value	122260
href	24504
note	19
cved	47
deprecated	1415
negate	48
cvss	33
check	55
publishedDatetime	47
reference	24449
operator	48
product	3760
lastModifiedDatetime	47
system	55
source	126
factRef	3762
notes	19

Table 4 - Extracted Relations from Structured CVE 2014 Data Feeds

As it can be seen from the table above, there are thirty different relation type extracted from the CVE which is very low compared to the relation types indicated in corresponding XSD data. This is mainly caused by the lack of diversification of the data which is entered to this feed. Hopefully it will be increased within the data feed in subsequent years. As the diversity of relation types are increased, more structured relation could be extracted and ontology will be enriched in the sense of both quality and coverage.

4.2 Evaluation of Knowledge Base from Unstructured Data

The second phase is the evaluation of knowledge extraction from unstructured sources which is the bottleneck for the overall success of the system. Performance measurement methodologies are put forward by the common Information Retrieval tasks. "Precision" and "Recall" are the two metrics for the evaluation of such systems. Precision is the proportion of the correctly identified items to the total number of identified items whereas recall value denotes the proportion of the number of correctly identified items to the total number of correct items that are available within the task. These two measurements can be generalized with the following formulas [4];

$$\text{Precision} = \frac{|\{\text{Relevant}\} \cap \{\text{Retrieved}\}|}{|\{\text{Retrieved}\}|}$$

$$\text{Recall} = \frac{|\{\text{Relevant}\} \cap \{\text{Retrieved}\}|}{|\{\text{Relevant}\}|}$$

Equation 5 - Precision and Recall Calculations

The boundaries of the resulting values of both Precision and Recall are 0 or 1. In order to analyze the overall test accuracy another measurement is used which is called "F-Measure" or sometimes denotes as "F1" score. The F1 score can be interpreted as a weighted average of the precision and recall, where an F1 score reaches its best value at 1 and worst score at 0. The evaluation of this value is indicated as follows [4];

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

Equation 6 - F-Measure Calculation Formula

These measurements are valid for knowledge assessment, however not sufficient for the evaluation from the perspective of the correctness and relevance for an ontological structure. A trusted reference knowledge base is required for the system evaluation. Since no standard reference knowledge base exist for this specific domain, a set should be constructed which includes manual annotations performed by humans. These manually annotated sets are called as "*Golden Set*", since the expected accurate results are included for a particular part of the given input set. [39] Defines a set of measurement which calculates the precision of the coverage with separate formulas in conjunction with the correctness and relevance. The measurements are formulized as follows;

$$Pr(\text{Correctness}) = \frac{N_j(\text{correct}) + N_j(\text{irrelevant})}{N_j(\text{correct}) + N_j(\text{incorrect}) + N_j(\text{irrelevant})}$$

$$Pr \begin{pmatrix} \text{Correctness} \\ + \\ \text{Relevance} \end{pmatrix} = \frac{N_j(\text{correct})}{N_j(\text{correct}) + N_j(\text{incorrect}) + N_j(\text{irrelevant})}$$

$$Cvg(\text{Correctness}) = \frac{N_j(\text{correct}) + N_j(\text{irrelevant})}{N_g(\text{correct}) + N_g(\text{irrelevant}) + N_g(\text{added})}$$

$$Cvg \begin{pmatrix} \text{Correctness} \\ + \\ \text{Relevance} \end{pmatrix} = \frac{N_j(\text{correct})}{N_g(\text{correct}) + N_g(\text{added})}$$

Equation 7 - Precision and Coverage Calculations with Correctness and Relevance Perspectives

N_j denotes for the number of the concept or relation which are identified by the proposed system in this study whereas N_g denotes the manually annotated items within the golden set. Indications are defined as follow; (correct) is the correctly extracted items, (irrelevant) is the items extracted but irrelevant to the Cyber Security domain, (incorrect) denotes the wrong extracted items and (added) refers to the items that are discarded by the proposed system but exists within the gold set. This methodology is used for the evaluation of the second phase.

Initially, a particular set from each Cyber Security data type is selected and manually annotated in order to generate the golden set which will be the reference point for

calculations. Then the same input is used for the system evaluation and resulting values are calculated for different set parameters. for each source type. As it mentioned in the NLP oriented data extraction module the statistical document measurements are parameterized to evaluate the proposed system's performance empirically. In addition to the statistical measurements the weight of Wikipedia Index usage within scoring process and the concept matching between DBpedia resources are also evaluated by incorporation of their scoring mechanisms to the existing parametric scoring system.

Three parameter set is defined for the evaluation phase. First set is called the Wiki based scoring in which the TFIDF parameter is favored in order to indicate the contribution of the usage of Wikipedia Indexes. Second set is defined to emphasize the effect of the data augmentation process which is based on the mapping of DBpedia resource graph. In the third set, the TFIDF and Graph scores are minimized in order to indicate the contribution of the primitive term statistical measurements. Independent from these three parameter set, the contribution of the base methodologies of the system can be observed in each result. Distributions of the parameters are indicated in the following tables for each parameter set. Not all of the statistical measurements defined in the NLP oriented information extraction module is incorporated to the parameter sets for the brevity.

Index Based Parameter Set	
firstOccurrenceOfEarliestToken	15%
firstOccurrenceOfPhrase	5%
graphScore	15%
phraseFreq	10%
relativeLength	5%
tfIdf	40%
wordCount	10%

Table 5 - Index Based Parameter Set for System Evaluation

Graph Based Parameter Set	
firstOccurrenceOfEarliestToken	10%
firstOccurrenceOfPhrase	5%
graphScore	45%
phraseFreq	10%

relativeLength	5%
tfIdf	15%
wordCount	10%

Table 6 - Graph Based Parameter Set for System Evaluation

Primitive Statistical Term Measurement Set	
firstOccurrenceOfEarliestToken	25%
firstOccurrenceOfPhrase	5%
graphScore	10%
phraseFreq	35%
relativeLength	5%
tfIdf	10%
wordCount	10%

Table 7 - Primitive Statistical Term Measurement Set for System Evaluation

These parameter sets are adjusted for the system and the resulting concepts, relations are compared with the reference annotated knowledge base. In order to perform a test bench is developed which calculates the measurements for ontology evaluation. The resulting set contains 6 different values for each of the parameter set in specified data source type which totally makes 54 values to evaluate the overall system performance from different point of views.

CyberSecurity Data Source Type	Precision		Coverage		F-Measure	
	Correctness	Correctness + Relevance	Correctness	Correctness + Relevance	Correctness	Correctness + Relevance
CVE	0.6190476 4	0.4761904 8	0.325	0.3030303	0.4262295 6	0.3703703
CCE	0.5714286	0.4285714 3	0.3	0.2727272 8	0.3934426 6	0.3333333 4
VVS	0.6086956	0.3913043 4	0.28	0.2093023 2	0.3835616 4	0.2727272 8

Table 8 - Evaluation Results of Index Scoring Based Parameter Set

CyberSecurity Data Source Type	Precision		Coverage		F-Measure	
	Correctness ss	Correctness ss + Relevance	Correctness ss	Correctness ss + Relevance	Correctness ss	Correctness ss + Relevance
CVE	0.5135135	0.3243243	0.5757575 6	0.4615384 6	0.5428571	0.3809523 6
CCE	0.5945946	0.3581081	0.5623003	0.436214	0.5779967	0.3933209 5
VVS	0.5945946	0.2905405 5	0.5415385	0.3372549	0.5668277	0.3121597 2

Table 9 - Evaluation Results of Graph Based Parameter Set

CyberSecurity Data Source Type	Precision		Coverage		F-Measure	
	Correctness ss	Correctness ss + Relevance	Correctness ss	Correctness ss + Relevance	Correctness ss	Correctness ss + Relevance
CVE	0.5333333 6	0.2	0.7529412 7	0.4411764 4	0.6243902	0.2752293 6
CCE	0.5882353 4	0.2205882	0.7619048	0.5514706 3	0.6639004	0.3151260 3
VVS	0.5333333 6	0.1833333 4	0.6464646 5	0.3767123 2	0.5844749	0.2466367 6

Table 10 - Evaluation Results of Primitive Statistical Term Measurement Set

Results of index scoring based system indicates good performance in precision as expected, since the utilization of Wiki Indexes contributes significantly to the extraction of correct concepts. On the other hand the coverage results slightly underperformed with this parameter set, since the weight of the graph scoring which is 15% does not influence the general calculation. By considering the low coverage results, it can be clearly stated that the

coverage of the resulting ontology is highly affected by the index scoring system which provides the elimination of the less significant concepts within the extraction phase. This theory can also be verified by observing the results of the second table in which the graph score parameter weight is emphasized. The results of coverage values are balanced and gets closer to the precision values. The dramatic change in terms of coverage can be observed in the third parameter set in which both weights of the graph and index parameters are lowered to provide a homogenous distribution. With the combination of these parameter weight distributions, the coverage value is increased in a good way whereas the relevance values are decreased in a serious portion which is also an expected result and verifies the contribution of the both of the scoring methodologies index and graph. As a consequence of this system evaluation, the coverage of the ontology can be increased by lowering the scoring mechanism which eliminates the domain irrelevant concepts from the given source. However the main purpose of this study is to generate a refined ontology for which the relevancy and correctness of the knowledge is more significant in the sense of cyber security system automation idea.

CHAPTER 5

CONCLUSION

Conclusion part of the study is separated under three parts which are the contribution of this study to the science and other researchers in this or related topics, discussion

5.1 Contribution & Discussion

This study is directed towards to filling the gaps in the automated cyber security systems evolution. After a deep examination of the both manual and semi-automatic works done, the purpose of the study is shaped among the idea of automated knowledge generation. All of the modules that are designated in the scope of this work have their own contributions to this purpose. This study as a whole can be regarded as an intelligent agent who produces knowledge base by plugging into the existing cyber security information sources. The contribution of each module is stated in individual sections.

Input Data Collection

Common Vulnerability Enumeration of NVD is a widely used cyber security related structured data. There exist various systems that utilized from this information. The usage of NVD data is different in the sense of connecting different enumerations types in a compound manner. CPE and CCE are incorporated to the input collector module so that product and vulnerable configurations that are referenced in CVE can be explicitly accommodated in the resulting knowledge base. This provides a full tracing of data in

analyze cyber security risks. Additionally, VVS data contributes to the enhancement of the knowledge by indicating the vendor statements for specific vulnerabilities and product versions. This study not only generates knowledge about vulnerabilities, but also enhances data with the product details, configuration details and vendor statements which ease the burden of security experts to seek information within different databases or on web. This is one of the novel approaches stated within the scope of this study.

RDF Generation From Relational Objects

There exist tools for converting XML to RDF data. XSLT (Extensible Stylesheet Language Transformations) is usually used for this transformation operation. This methodology is used in different domains from a technical perspective, however in the case of NVD data sources; this methodology is not preferred by researchers. An argument about this issue is clearly stated in the master thesis study of [\[17\]](#). "*NVD has deep nesting and similar names for properties, for instance "source" can be both vulnerability source and severity score source*". The reason why they manually modeled their ontology is defended with this argument. However, in the scope of this module, the claimed problems are addressed in the relational object mapping process. Each conflicted namespace is resolved in with the JAXB binding configurations and also "@XMLElement(name = xyz)" annotation is added for each object relation by the framework automatically which distinguishes same named relations. The binding part is done manually, however this is not a big deal from the perspective of the knowledge base generation and can be accepted as the input source refinement issue which does not disturbs the generality of the module. The consequence of this module is automatically generated RDF triples which constitute a base for the ontology in terms of both coverage and precision. One of the unique features of this module is the externalization of the application logic which provides the reusability of the processes with different relational object sources which may be added in subsequently or modified day by day.

Static ontology models strictly bounds to the data source schemas which decrease their usability in running systems. Majority of these kinds of systems remains at the proof of concept state and cannot be fully utilized by the end users without any contribution from security experts. Contemporarily, there is a tendency to quit from static models. Since, schemas and data types can be changed by time. This is one of the biggest IT problems

around the world. This study gives a lead for the dynamic modeled knowledge bases in cyber security domain by excluding the manual operations and statically modeling as much as possible. This is the reason why this approach is persistently applied. It is an undeniable fact that the automatically generated RDF triples by this module is not very useful in the sense of analyzing threats, however the point here is that to lay the foundations of the knowledge base for the preceding processes performed in the next module which is NLP oriented knowledge extraction.

NLP Oriented Knowledge Extraction

Usually, knowledge generation projects are conducted by manually accumulating the knowledge which belongs to different people. Web interfaces are provided for this issue in order to facilitate the flow of data easily from users to the knowledge base. Taxonomical information and related links are specified for each added concept by the corresponding user. To automate this process is also one of the major tasks of this study. This module is the core part of the whole system since this is the actual phase in which the knowledge is extracted from the given source. The hidden knowledge implications are exists within the given inputs that should be detected and semantically combined in an RDF format for the knowledge base. The data source candidates for the knowledge extraction are mainly the human languages because the only way to transfer knowledge without using any machine readable structure is to using human language. This is the reason why there are vulnerability description and notes sections in CVE, configuration details in CCE and statements in VVS. All of these data is unstructured and accommodate the knowledge with itself. Security experts are utilized from these data in decision making and reasoning. NLP oriented approaches are the key factor for automating decision making and reasoning tasks. Therefore, to extract semantic concepts including relations from given texts are the indispensable part of this study. To facilitate such a process the biggest step for the knowledge base which will be used by intelligent systems? From the perspective of this study, the foundations of creating an intelligent autonomous cyber security system lays down on the usage of an intelligently generated knowledge base which is the aim of this study.

Semantic parsing is applied in numerous researches, especially in the study of [8] sophisticated semantic frame parsing methodologies are used and tested. In the scope of

the NLP oriented knowledge extractor, a novel semantic parsing method is tried to apply with a sequence of operations like chunking and POS-Tagging. The overall system evaluation is performed, whereas no special evaluation methodology is included in the scope of this work. However the general results of extracted concepts are promising although the sole purpose of the system is outperforming the current research related to this task. One of the most important processes that contribute to the overall knowledge extraction phase is the phrase scoring part.

With the combination of text based and graph based term/document scoring methodologies, a novel approach is put forward by this study. Selected parameters are uniquely weighted for the scoring process and tested with the same reference test bench in order to provide a comparison of their contribution to the overall success in terms of both relevance and precision. Coverage is one of the main issues in ontology development, however relevancy factor is slight more significant within the scope of this study. Since the generated knowledge is desired to use by the cyber security automation systems for which the relevancy of the knowledge is important in the sense of decision making. Irrelevant relations and concepts may decrease the quality of the queried data in terms of the ambiguity among the results. An extra elimination operation would be required in such cases which is not a desired task.

Incorporation of both Wikipedia index and Freebase graph based utilization to the scoring system are highly contributed to the overall success of the proposed system and gives a unique idea about the utilization of the external data in knowledge generation processes. The power of ontology enhancement is mainly taken from this part of the module which can be also used in other knowledge construction problems in domain rather than cyber security.

As a consequence, the resulting knowledge base of this study can be utilized from both intelligent software agents and also by human security experts. It helps to seek the relations between vulnerabilities, threats, software configurations and counter measures. Especially for zero-day attacks that constitutes risks for the companies [40]. The signature of the attack may not be officially mentioned in the records, however based on this generated knowledge an implicit association can be revealed with a new input threat and configuration type which may direct the security experts or systems to take necessary precautions.

5.2 Limitations & Future Research

Proposed approaches and implementations in this study give an idea for the solutions that addresses to the defined cyber security automation problems. In each module of the system there are particular fields that can be improved and supported. These fields are examined from two different perspectives which are the internal improvements to the core system modules by redesign them and the external improvements like changing input data sources or externally utilized data usage.

Internal improvements can be mainly considered for the NLP oriented extractor module. Especially methodologies used in semantic parsing may be fortified and stabilized by using more sophisticated NLP processes including supervised or unsupervised learning algorithms. In parallel to the researches in NLP fields, the corresponding parts of module can be redesigned in order to improve the correctness of part of speech detection from the unstructured texts. However within the current limitations of the NLP methodologies, there exist no methodology that is able to completely parse the given sentence and extracts the same results with the gold annotation set which is performed by human experts. It is an undeniable fact that there exists no AI that can perform most of the tasks belongs to the human brain including capability of association. Another aspect for this module is related to the *Word Sense Disambiguation* which is one of the significant parts of the named entity recognition problem. In the phrase scoring part of this module, while matching the concepts with the concepts that are exists in external sources, Word Sense Disambiguation task should be performed to differentiate the implicit senses of the concepts in order not to confuse between the meanings of the concepts which can have different meaning in different contexts. The concept "*apple*" can be given as example which may refer to the corporation Apple Inc. or the fruit. This part is not addressed in the scope of this study and may be a future work which contributes to the enhancement of the knowledge base.

As it is mentioned in the initial part of the proposed system methodologies, the selection of the input data sources is very critical in terms of both precision and the relevance of the generated knowledge base. Therefore, input data pieces are sensitively selected and refined in a standalone module. One of the advantageous parts of this study is the low

cohesion between modules. This facilitates the modification of inputs or gives opportunity to completely change of input sources which highly changes the overall success of the system. There is possibility of creation of new structured data In addition to the CVE, VVS, CCE and CPE. The new data sources may include more details about vulnerabilities, threats and countermeasures. The usage of the newly discovered data is possible within the scope of this study. On the other hand, external data utilization part can also be substituted with other corpora rather than Wikipedia, Freebase or DBpedia. At this point the only limitation for the knowledge base is that it is static RDF representation which cannot be changed externally.

Although the study seems complete for the defined research topic, from the perspective of the system this work only constitutes the base ground for the imagined system which is capable of decision making and reasoning. The generated knowledge base in this study would be meaningless unless it is used by an intelligent cyber security automation system. While designing and implementing this system, all struggles was given for considering this complementary part of the study. To confide in security automation in detection and prevention phases, a trusted knowledge base should be incorporated to the whole system.

REFERENCES

- [1] Vorobiev, A., & Bekmamedova, N. An Ontology-Driven Approach Applied to Information Security. *Journal of Research and Practice in Information Technology*, 42 (1), pp. 61-76, February 2010
- [2] Herzog, A., Shahmehri, N., & Duma, C. (n.d.). An Ontology of Information Security. *International Journal of Information Security and Privacy*, 1-23. Retrieved January 30, 2015, from <http://www.igi-global.com/article/ontology-information-security/2468c>
- [3] Takahashi, T., and Kadobayashi, Y. Cybersecurity Information Exchange Techniques: Cybersecurity Information Ontology and CYBEX. *Journal of the National Institute of Information and Communications Technology* (2010), 127–136.
- [4] Wimalasuriya, D., & Dou, D. (2010). Ontology-based information extraction: An introduction and a survey of current approaches. *Journal of Information Science*, 306-323.
- [5] Undercoffer, J., Joshi, A., and Pinkston, J. 2003. Modeling computer attacks: An ontology for intrusion detection. In *Proceedings of the 6th International Symposium on Recent Advances in Intrusion Detection (RAID'03)*. Pittsburgh, PA. Lecture Notes in Computer Science, vol. 2820, 113–135.
- [6] AHMAD, K. and GILLAM, L. (2005). Automatic ontology extraction from unstructured texts. In *On the Move to Meaningful Internet Systems – OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2005*, Lecture Notes in Computer Science 3761, R. Meersman and Z. Tari, Eds., Springer, pp. 1330–1346.
- [7] M. Balakrishna and M. Srikanth. 2008. Automatic ontology creation from text for national intelligence priorities framework (NIPF). In *Proceedings of 3rd International Ontology for the Intelligence Community (OIC) Conference*, pages 8–12, Fairfax, VA, USA, December 3-4
- [8] I. Bedini and B. Nguyen. "Automatic ontology generation: State of the art". In: PRISM Laboratory Technical Report. University of Versailles (2007). 13- I. Bedini and B. Nguyen. "Automatic ontology generation: State of the art". In: PRISM Laboratory Technical Report. University of Versailles (2007).
- [9] Wali, A., Chun, S. A., and Geller, J. "A Bootstrapping Approach for Developing a Cyber-Security Ontology Using Textbook Index Terms", *International Conference on Availability, Reliability and Security*, 2013.
- [10] Kotenko, I., Polubelova, O., Chechulin, A., & Saenko, I. (n.d.). Design and Implementation of a Hybrid Ontological-Relational Data Repository for SIEM Systems. *Future Internet*, 355-375.

- [11] Fenz, T. Pruckner, and A. Manutscheri, "Ontological mapping of information security best-practice guidelines," in Business Information Systems, ser. Lecture Notes in Business Information Processing, W. Abramowicz, Ed. Heidelberg: Springer, 2009, vol. 21, pp. 49–60.
- [12] Nguyen, V.: Ontologies and Information Systems: A Literature Survey (2011)[17]
Nguyen, V.: Ontologies and Information Systems: A Literature Survey (2011)
- [13] G. Elahi, E. S. K. Yu, and N. Zannone, "A modeling ontology for integrating vulnerabilities into security requirements conceptual foundations," in ER, ser. Lecture Notes in Computer Science, A. H. F. Laender, S. Castano, U. Dayal, F. Casati, and J. P. M. de Oliveira, Eds., vol. 5829. Springer, 2009, pp. 99-114.
- [14] J. A. Wang and M. Guo. OVM: An Ontology for Vulnerability Management. In Proceedings of the 5th Annual Workshop on Cyber Security and Information Intelligence Research: Cyber Security and Information Intelligence Challenges and Strategies (CSIIRW'09), pages 34:1-34:4, New York, NY, USA, 2009. ACM.
- [15] J.A. Wang, M.Z. Guo, "Security data mining in an ontology for vulnerability,", Proceedings of 2009 International Joint Conference on Bioinformatics, Systems Biology and Intelligent Computing, pp. 597- 603, 2009
- [16] T. Takahashi, Y. Kadobayashi and H. Fujiwara, "Ontological Approach toward Cybersecurity in Cloud Computing", 2010
- [17] A. Joshi, R. Lal, T. Finin, and A. Joshi. Extracting cybersecurity related linked data from text. In Semantic Computing (ICSC), 2013 IEEE Seventh International Conference on, pages 252–259. IEEE, 2013
- [18] Urbanska, Malgorzata (2014). Automated security analysis of the home computer (M.S. dissertation COLORADO STATE UNIVERSITY). Retrieved from <http://gradworks.umi.com/15/58/1558327.html>
- [19] Varish Mulwad, Wenjia Li, Anupam Joshi, Tim Finin, and Krishnamurthy Viswanathan, Extracting Information about Security Vulnerabilities from Web Text. Web Intelligence/IAT Workshops, page 257-260. IEEE Computer Society, (2011)
- [20] R. A. Bridges, C. L. Jones, M. D. Iannacone, K. M. Testa, J. R. Goodall. "Automatic Labeling for Entity Extraction in Cyber Security". ASE Third International Conference on Cyber Security, Stanford, CA, May 27-31, 2014. (PTS#: 48680)
- [21] Turney, P.D.: Learning Algorithms for Keyphrase Extraction. Information Retrieval, 2 (2000) 303-336
- [22] CHANG, K. (2014, June 02). Automating Cybersecurity. Retrieved 01 26, 2015, from http://www.nytimes.com/2014/06/03/science/automating-cybersecurity.html?_r=1
- [23] Jackson, W. (2014, July 25). CYBEREYE. Retrieved 01 26, 2015, from GCN Blogs: <http://gcn.com/blogs/cybereye/2014/07/humans-vs-automation.aspx>

- [24] Lydon, B. (2015, January 26). Avoiding Cyber Security Disasters. Retrieved 01 26, 2015, from <http://www.automation.com/portals/manufacturing-operations-management/cyber-security/avoiding-cyber-security-disasters>
- [25] P. Buitelaar, P. Cimiano and B. Magnini, "Ontology Learning from Text: An Overview," *Ontology Learning from Text: Methods, Evaluation and Applications*, P. Buitelaar, P. Cimiano, and B. Magnini, eds., vol. 123, *Frontiers in Artificial Intelligence and Applications*, pp. 3-12, IOS Press, 2005.
- [26] Dempsey K., W. G. (February 19, 2014). Summary of NIST SP 800-53 Revision 4, Security and Privacy Controls for Federal Information Systems and Organizations. U.S.: NIST Computer Security Division.
- [27] J. Malefant, M. Jacques, and F.N. Demers. A Tutorial on Behavioral Reflection and its Implementation. In Proceedings of REFLECTION '96. ECOOP, 1996
- [28] Alex Franz, T. B. (2006, August 03). All Our N-gram are Belong to You. Retrieved 01 26, 2015, from Google Research Blog: <http://googleresearch.blogspot.com.tr/2006/08/all-our-n-gram-are-belong-to-you.html>
- [29] Santorini, B. (June 1990). Part-of-Speech Tagging Guidelines for the Penn Treebasnk Project. Pennsylvania: Department of Computer and Information Science, University of Pennsylvania.
- [30] Rajaraman, A., Ullman, J. D. (2012). *Mining of massive datasets*. Cambridge: Cambridge University Press. ISBN: 9781139157926 1139157922 9781107015357 1107015359
- [31] Bill Tancer (May 1, 2007). "Look Who's Using Wikipedia". Time. Retrieved December 1, 2007. The sheer volume of content [...] is partly responsible for the site's dominance as an online reference. When compared to the top 3,200 educational reference sites in the US, Wikipedia is No. 1, capturing 24.3% of all visits to the category. Cf Bill Tancer (Global Manager, Hitwise), "Wikipedia, Search and School Homework", Hitwise, March 1, 2007.
- [32] Pablo Mendes, Max Jakob, Andrés García-Silva and Christian Bizer. DBpedia Spotlight: Shedding Light on the Web of Documents. In the Proceedings of the 7th International Conference on Semantic Systems (I-Semantics). Graz, Austria, 7–9 September 2011
- [33] Markoff, J. (2007-03-09). "Start-up Aims for Database to Automate Web Searching" . The New York Times. Retrieved 2007-03-09.
- [34] Freebase. (2015, January 28). Freebase Data Dumps. Retrieved 01 26, 2015, from Freebase API Data Dumps: <https://developers.google.com/freebase/data>
- [35] Beckett, D. (2014). N-Triples W3C RDF Core WG Internal Working Draft. Retrieved 01 22, 2015, from <http://www.w3.org/2001/sw/RDFCore/ntriples/>
- [36] Shellac. (2011, 29 October). JUNG wrapper for jena models. Retrieved 01 16, 2015, from <https://github.com/shellac/JenaJung>

[37] "Facts about Google and Competition" . Archived from the original on 4 November 2011. Retrieved 12 July 2014.

[38] Sobek, M. The PageRank Algorithm. Retrieved 01 22 2015, from <http://pr.efactory.de/e-pagerank-algorithm.shtml>

[39] A. Gangemi, C. Catenacci, M. Ciaramita, and J. Lehmann, "Modeling ontology evaluation and validation" in European Semantic Web Symposium/Conference (ESWC), 2006, pp. 140-154

[40] Joshi, A. How to protect your company from 'zero-day' exploits. Retrieved 01 24 2015, from <http://www.computerworld.com/article/2575145/security0/how-to-protect-your-company-from--zero-day--exploits.html>

[41] Mike Schiffman, Gerhard Eschelbeck, David Ahmad, Andrew Wright, Sasha Romanosky, "CVSS: A Common Vulnerability Scoring System", National Infrastructure Advisory Council (NIAC), 2004.

[42] Bizer, Christian; Lehmann, Jens; Kobilarov, Georgi; Auer, Soren; Becker, Christian; Cyganiak, Richard; Hellmann, Sebastian (September 2009). "DBpedia - A crystallization point for the Web of Data" . Web Semantics: Science, Services and Agents on the World Wide Web 7 (3): 154–165

[43] Markoff, John (2007-03-09). "Start-up Aims for Database to Automate Web Searching" . The New York Times. Retrieved 01 16, 2015

APPENDICES

APPENDIX A

TRIPLE GENERATOR ALGORITHM HELPER METHODS

```
private ArrayList<MyField> getAllFields(Object obj)
{
    Field[] fields = obj.getClass().getDeclaredFields();
    ArrayList<MyField> fieldArrayList = ConvertFieldsToMyFields(fields, obj);

    Class<?> superClass = obj.getClass().getSuperclass();

    if (superClass.getSimpleName().equals("Object"))
    {
        superClass = null;
    }

    while (superClass != null)
    {
        Field[] fieldsOfSuperClass = superClass.getDeclaredFields();
        ArrayList<MyField> myFields = ConvertFieldsToMyFields(fieldsOfSuperClass, obj);

        //Union of arrays
        for (int i = 0; i < myFields.size(); i++)
        {
            fieldArrayList.add(myFields.get(i));
        }

        superClass = superClass.getClass().getSuperclass();
        if (superClass.getSimpleName().equals("Object"))
        {
            superClass = null;
        }
    }

    return fieldArrayList;
}
```

Figure 36 - Helper Method for Retrieving All Fields In Object

```

private ArrayList<MyField> ConvertFieldsToMyFields(Field[] fields, Object obj)
{
    ArrayList<MyField> objs = new ArrayList<MyField>();
    for (int i = 0; i < fields.length; i++)
    {
        fields[i].setAccessible(true); // if you want to modify private or protected fields
        try {
            if (fields[i].get(obj) == null || fields[i].getModifiers() != 4)
            {
                continue;
            }
        } catch (IllegalArgumentException e1)
        {
            continue;
        } catch (IllegalAccessException e1)
        {
            continue;
        }
    }

    MyField myField = new MyField();
    myField.obj = null;
    try {
        myField.obj = fields[i].get(obj);
    } catch (IllegalArgumentException e) {
    } catch (IllegalAccessException e) {
    }
    myField.name = fields[i].getName();
    myField.classType = fields[i].getType();

    if (myField.obj != null)
    {
        objs.add(myField);
    }
}
return objs;
}

```

Figure 37 - Helper Method for Field Conversion

```

private boolean isJAXBEntity(Object o)
{
    boolean isAnnotationExists = o.getClass().isAnnotationPresent(XmlAccessorType.class);
    return isAnnotationExists;
}

private Resource addResourceToRDFResource(Resource r1, String propName, Resource value)
{
    Property p = model.createProperty(NS + propName);
    Resource resultingNode = r1.addProperty(p, value);
    return resultingNode;
}

private Resource createInheritRDFResource(String typeName)
{
    Resource r = model.createResource();
    Property p = model.createProperty(NS + inheritPropertyName);
    Resource resultingNode = r.addProperty(p, typeName);
    return r;
}

private void createRDFResource(Resource resource, String property, String fieldValue)
{
    Property p = model.createProperty(NS + property);
    resource.addProperty(p, fieldValue);
}

private Property createRDFProperty(String property)
{
    Property p = model.createProperty(NS + property);
    return p;
}

```

Figure 38 - Helper Methods for RDF Model Population

```

private void printStatements()
{
    // list the statements in the Model
    StmtIterator iter = model.listStatements();

    // print out the predicate, subject and object of each statement
    while (iter.hasNext()) {
        Statement stmt      = iter.nextStatement(); // get next statement
        Resource  subject   = stmt.getSubject();      // get the subject
        Property  predicate = stmt.getPredicate();    // get the predicate
        RDFNode   object    = stmt.getObject();       // get the object

        System.out.print(subject.toString());
        System.out.print(" " + predicate.toString() + " ");
        if (object instanceof Resource) {
            System.out.print(object.toString());
        } else {
            // object is a literal
            System.out.print(" \"" + object.toString() + "\"");
        }

        System.out.println(" .");
    }
}

```

Figure 39 - Helper Method for Printing RDF Statements

APPENDIX B

DBpedia ONTOLOGY

DBpedia is a project aiming to extract structured information from the information created as part of the Wikipedia project and then to make this information available on the web. DBpedia allows user to query relationships and properties associated with Wikipedia resources. [42]



DBpedia Ontology is a shallow, cross-domain ontology, which has been manually created based on the most commonly used info boxes within Wikipedia. The ontology currently covers over 259 classes which form a subsumption hierarchy and are described by 1,200 different properties.

The ontology is instantiated by an info box data extraction method which is based on hand-generated mappings of Wikipedia info boxes to the DBpedia ontology. The mappings define fine-granular rules on how to parse info box values. The externally utilized sources by DBpedia are indicated in the following figure;

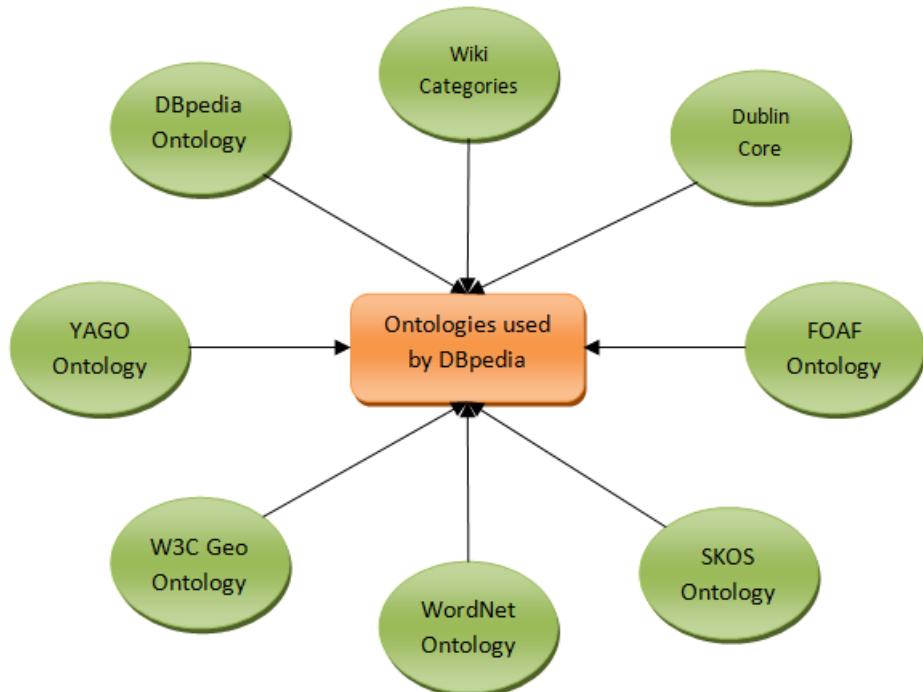


Figure 40 - Ontologies Used by DBpedia

APPENDIX C

FREEBASE ONTOLOGY

Freebase is a large collaborative knowledge base. It is an online collection of structured data harvested from many sources, including individual 'wiki' contribution. Freebase aims to create a global resource which allows people (and machines) to access common information more effectively. It is developed by the American software company Metaweb and has been running publicly since March 2007. [\[43\]](#)



As a remark, Freebase project is decided to transfer to Wikidata project starting by the middle of 2015 and they will launch a new API for entity search powered by Google's Knowledge Graph.

Freebase contains more than 10 million topics, more than 3000 types, and more than 30,000 properties. The types used in Freebase are grouped into 75 different domains according to their related category. Each domain contains a set of types and each type has previously defined properties. Also all of the domains, types, and properties are defined with a unique id that enables accessing and using data sets accurately.

Every topic page that is seen on Freebase web site is marked with a unique URL defining the corresponding topic. This approach brings benefits when defining different topics with the same keyword. For instance, "Bob Dylan" can be an instance of different types: song writer, singer performer, book author, film actor etc. So, for every type of the same concept there is a corresponding topic page and also unique topic page URL.

Americanfootball | Architecture | Astronomy | Automotive | Aviation | Awards | Baseball | Basketball | Bicycles | Biology | Boats | Books | Broadcast | Business | Celebrities | Chemistry | Comics | Common | Computers | ConferencesandConventions | Cricket | DataWorld | Dicicams | Education | Engineering | Event | Fashion,ClothingandTextiles | FictionalUniverses | Film | Food&Drink | Freebase | Games | Geology | Government | HobbiesandInterests | IceHockey | Influence | Internet | Language | Law | Library | Location | MartialArts | MeasurementUnit | Media | Medicine | MetawebSystemTypes | Meteorology | Military | Music | Olympics | Opera | Organization | People | Periodicals | PhysicalGeography | Physics | Projects | ProtectedPlaces | Radio | Rail | Religion | RoyaltyandNobility | Soccer | Spaceflight | Sports | Symbols | Tennis | Theater | Time | Transportation | Travel | TV | VideoGames | VisualArt |

[Figure 41 - Freebase Domains](#)

APPENDIX D

APACHE SolR

Apache Solr is a high performance search server built using Apache Lucene. Its major features include full-text search, hit highlighting, faceted search, near real-time indexing, dynamic clustering, database integration, rich document (e.g., Word, PDF) handling, geospatial search, and a web admin interface. It is highly reliable, scalable and fault tolerant, providing distributed indexing, replication and load-balanced querying, centralized configuration, automated failover and recovery.

SolR has XML/HTTP and JSON APIs, making it easy to use from any programming language. Currently there are APIs for Java, .Net, Ruby, Rails, Python, PHP, Perl, AJAX, and Javascript. It also has a powerful external configuration adjusting most applications without any Java coding, and it has extensive plugin architecture when more customization is needed.

Apache SolR is a search server with a REST-like API. SolR indexes data via XML, JSON, CSV or binary over HTTP and queries results via HTTP GET.

The Data Import Handler of SolR provides a mechanism for importing data from a data store and indexing it. These data stores can be databases (e.g., MySQL, MS SQL, MongoDB), HTTP based data sources like RSS and ATOM feeds, email repositories and structured XML where an XPATH processor is used to generate fields. In order to use the Data Import Handler, a request handler must be added to the solrconfig.xml file.

```
<requestHandler name="/dataimport" class="org.apache.solr.handler.dataimport.DataImportHandler">
  <lst name="defaults">
    <str name="config">data-config.xml</str>
  </lst>
</requestHandler>
```

In the data-config.xml file, data sources and their respective mappings are configured. The data source is specified giving the type, driver and jdbc url of the database instance to be imported.

```
<dataSource type="JdbcDataSource" driver="com.mysql.jdbc.Driver" url="jdbc:mysql://localhost/classicModels"
  user="user" password="password" />
```

Along with the data source, the Data Import Handler also requires an entity processor to extract data from the data source, transform it and add it to the SolR index. As can be seen from the entity processor below, the database fields are mapped to the schema variables of the SolR index.

```

<document name="products">
  <entity name="product" query="select * from products"
    deltaQuery="select productCode from products where last_modified > '${dataimporter.last_index_time}'">
    <field column="productCode" name="id" />
    <field column="productName" name="name" />
    <field column="productVendor" name="manu" />
    <field column="buyPrice" name="price" />
    <field column="productDescription" name="description" />
  </entity>
</document>

```

The Data Import Handler has two import modes; Full-import and Delta-import. When a full import operation is started, it will immediately start in a new thread and thus will not block any queries during the import. Full import stores the start time of the import in a file named dataimport.properties matching the query in the entity processor will be imported and indexed. Delta import is used for incremental imports and change detection. For delta import, the deltaQuery of entity processor is used, which selects the primary keys of the rows that will be updated.

Searches are done via HTTP GET in SolR using the SearchRequestHandler, which usually is configured in solrconfig.xml as “/select” for the path. SearchRequestHandler takes a number of parameters for the query some of which are;

- q: The query text.
- fl: Field list to be returned.
- sort: Sort specifications.
- wt: Response format(e.g., JSON, XML)
- hl.*: Specifications for highlighting terms in results.
- facet.*: Specifications for faceted search (generation of counts for categories/fields).

In order to query via SolrJ, first a SolrServer instance must be created. HttpSolrServer is used to connect to a Solr instance over the HTTP protocol which requires url address of the server. The HttpSolrServer also allows to change connection settings.

```

HttpSolrServer server = new HttpSolrServer("http://localhost:8983/solr");
server.setMaxRetries(1); // defaults to 0. > 1 not recommended.
server.setConnectionTimeout(5000); // 5 seconds to establish TCP
server.setDefaultMaxConnectionsPerHost(100);
server.setMaxTotalConnections(100);

```

The SolrQuery class contains all the properties of a query to be sent to the server like query, field list, highlighting etc. An instance of this class is sent to the server via the query method of SolrServer. To give more details about querying, below listed are some main methods of the SolrQuery class;

- setQuery(String query): Sets the text to be searched.
- setStart(int start): Sets the starting index of the results.
- setRows(int rows): Sets the number of results to be returned.

- addSort(String field, SolrQuery.ORDER order): Adds a single sort clause to the end of the current sort information.
- setFields(String... fields): Sets the field names to be returned in each result. E.g., in the example above, the results only contain “name” and “manu” information.
- setHighlight(Boolean b): Enables/disables highlighting.
- setFacet(Boolean b): Enables/disables facetting.

Here is a simple example returning the top 10 results sorted by their price.

which will later be used for delta import. During this import, all entities

```
SolrServer server = new HttpSolrServer(url);

SolrQuery query = new SolrQuery("Ford");
query.addSort("price", ORDER.desc);
query.setFields("name", "manu");
query.setRows(10);

QueryResponse response = server.query(query);

SolrDocumentList resultList = response.getResults();

System.out.println("Names of the top 10 results containing 'Ford' sorted by price:");

for (SolrDocument result : resultList) {
    System.out.println(result.getFieldValue("name"));
}
```

Solr has a web admin interface which enables users to easily view configuration details, run queries and arrange solr configurations. Solr admin interface can be configured using the admin section of the solrconfig.xml file.

TEZ FOTOKOPİ İZİN FORMU

ENSTİTÜ

Fen Bilimleri Enstitüsü

Sosyal Bilimler Enstitüsü

Uygulamalı Matematik Enstitüsü

Enformatik Enstitüsü

Deniz Bilimleri Enstitüsü

YAZARIN

Soyadı : Hassucu

Adı : Alp Gökhan

Bölümü : Bilgisayar Sistemleri Yüksek Lisans

TEZİN ADI (İngilizce) : Semantic Concept Recognition from Structured and Unstructured Inputs Within Cyber Security Domain

TEZİN TÜRÜ : Yüksek Lisans

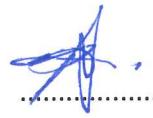


Doktora



1. Tezimin tamamı dünya çapında erişime açılsın ve kaynak gösterilmek şartıyla tezimin bir kısmı veya tamamının fotokopisi alınınsın.
2. Tezimin tamamı yalnızca Orta Doğu Teknik Üniversitesi kullancılarının erişimine açılsın. (Bu seçenekle tezinizin fotokopisi ya da elektronik kopyası Kütüphane aracılığı ile ODTÜ dışına dağıtılmayacaktır.)
3. Tezim bir (1) yıl süreyle erişime kapalı olsun. (Bu seçenekle tezinizin fotokopisi ya da elektronik kopyası Kütüphane aracılığı ile ODTÜ dışına dağıtılmayacaktır.)

Yazarın imzası



Tarih 05.03.2015