GENERATION OF CRIME KNOWLEDGE BASE FROM ONLINE NEWS ARTICLES

Thesis

Submitted in partial fulfilment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

by

Srinivasa K

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA
SURATHKAL, MANGALORE - 575 025
March, 2022

DECLARATION

by the Ph.D. Research Scholar

I hereby declare that the Research Thesis entitled **Generation of Crime Knowledge Base from Online News Articles** which is being submitted to the **National Institute of Technology Karnataka, Surathkal** in partial fulfilment of the requirements for the award of the Degree of **Doctor of Philosophy** in Department of Computer Science and Engineering is a bonafide report of the research work carried out by me. The material contained in this Research Thesis has not been submitted to any University or Institution for the award of any degree.

Srinivasa K, 177137CO006
Department of Computer Science and Engineering

Place: NITK, Surathkal.
Date: March 09, 2022

CERTIFICATE

This is to certify that the Research Thesis entitled **Generation of Crime Knowledge Base from Online News Articles** submitted by **Srinivasa K** (Register Number: 177137CO006) as the record of the research work carried out by him, is accepted as the Research Thesis submission in partial fulfilment of the requirements for the award of degree of **Doctor of Philosophy**.

10/03/2022

Dr. P. Santhi Thilagam
Research Guide
(Signature with Date and Seal)

Dr. P. SANTHI THILAGAM
Professor [redacted]
Department of Computer Science & Engineering
National Institute of Technology Karnataka, Surathkal
P.O. Srinivasnagar, MANGALORE, INDIA - 575025

10/03/22

Chairman - DRPC
(Signature with Date and Seal)

Chairman
~~DUGC~~ / ~~DPGC~~ / DRPC
Dept. of Computer Engg.
NITK - Surathkal
Srinivasnagar - 575 025

Dedicated to Sri Hari Vayu Gurugalu and My Family.

ACKNOWLEDGEMENTS

I take this opportunity to express my sincere gratitude to all the people who encouraged me to finish this dissertation. First and foremost, I would like to express my sincere gratitude to my research supervisor Dr.P.Santhi Thilagam, Professor, Department of Computer Science and Engineering, for constantly supporting me throughout the journey of my research with her knowledge, experience, and patience. I owe my deepest gratitude to her for the valuable guidance, timely suggestions, and advices. Beyond the subject area of this thesis, I have learnt a lot from her that, I am sure, will be useful in the later stages of my career as well as my life.

I would like to extend my gratitude to the members of my research progress assessment committee, Dr.Alwyn R. Pais and Dr.D.Pushparaj Shetty, for their insightful feedback and constructive suggestions throughout the period of my research work. I also extend my sincere thanks to the entire faculty of the Department of Computer Science and Engineering for rendering their help and support throughout the period of my research. Nevertheless, I am also grateful to the technical and administrative staff of the department for their timely help and co-operation to carry out the research work. I would like to express my sincere thanks to NITK for providing the necessary infrastructure and facilities required for the successful completion of this research work. I also express my sincere thanks to coordinator and all the staff of QIP cell of NITK, Dr.Islur, Mr.Shashidhar (Retired), Mr.Dayanand and Mr.Sarthak for their constant support and efforts in sanctioning our stipend from All India Council for Technical Education (AICTE).

I also extend my sincere gratitude to AICTE, Govt. of India for providing me the opportunity to do the research work under the Quality Improvement Programme of Govt. of India. Similarly, I also extend my heartiest gratitude to the management of my home institute Siddaganga Institute of Technology, Tumakuru for sponsoring and granting the leave for the whole journey of my Ph.D. work.

I would also like to extend my gratitude to my friends Dr. D.V.N. Sivakumar, Dr. Amit Praseed, Mrs. D. Umapriya, Mr. Subhash Acharya for their constant support and encouragement throughout the course of my research. I would like to express my gratitude to my other fellow doctoral students and friends for their healthy criticism and valuable co-operation. I also express my deepest gratitude to my friends and colleagues from my home institute who supported and encouraged me in all my difficult times during my Ph.D. work.

This thesis would not have been possible without the exceptional support of my family. I am indebted to my parents, my wife, my kids, my siblings family and my in-laws for their unconditional love, inspiration, encouragement, blessings, prayers and sacrifice over the years.

Finally, I would like to do a lot of pranams to Sri Hari, Vayu, Samasta Tatvika Devategalu and all Gurugalu for granting me everything needed to carry out my research work. Without their blessings nothing is possible.

Srinivasa K

ABSTRACT

The growing amount of unstructured data on the internet has piqued the interest of Semantic Web (SW) technology researchers in the creation of Knowledge Bases (KBs). A KB is a structured representation of unstructured data that can be read by machines. Facts are typically stored in the KB as a set of triples of the form (head entity, relation, tail entity), which represent the relationships between the head and the tail entities. In today's internet age, information about crime can be found in a variety of places, including news media, social networks, blogs, and video repositories, among others. Crime reports published in online newspapers are frequently regarded as more reliable than crowdsourced data such as that found on social media. Furthermore, information in newspapers is available in both multilingual text and image form. As a result, generating a KB of crime-related facts from online newspapers will be useful for Law Enforcement Agencies (LEAs) in analyzing crime activities without language and modality barriers. Furthermore, creating a KB from sources that publish data on a daily basis, such as news media, keeps the KB up to date.

The creation of a KB involves the extraction and integration of data from multiple sources. At the same time, it also ensures the accuracy of the extracted knowledge. Existing research has primarily focused on extracting entities and their relationships from mono-lingual sources, while ignoring the impact of extracted entities on generating a complete and non-redundant KB. Furthermore, the majority of them have used either corpus or knowledge-based similarity methods to integrate information from multiple sources without delving into the full semantics hidden in facts. These factors result in redundancy and the loss of critical information in the KB. In addition, the completion of a knowledge base necessitates an incremental update of the KB through the extraction of facts from multi-lingual sources. It is also critical to verify the credibility of facts before they are entered into the KB. To address the aforementioned issues, this study proposes a bootstrap-based model for developing Crime Base, a knowledge base of crime entities and their relationships that contains complete, non-redundant, and validated facts. It makes use of crime-related text and image data from English and Hindi online news articles.

To begin, the proposed model extracts crime-related facts from English news articles in order to construct the Crime Base. Unlike existing methods and tools, it extracts entities using an external KB-DBpedia with the goal of minimizing redundancy and loss of essential information. To capture more semantics during integration, a semantic merging method is proposed in which entities extracted from text data are correlated using both corpus and knowledge-based similarity measures, and image entities are correlated using both low-level and high-level image features. Empirical results show that using both similarity measures reduces redundancy in the KB more effectively than using either of the two.

Secondly, a clustering-based bootstrapping approach is proposed to enrich the Crime Base created with English news articles with Hindi news articles. The proposed method investigates redundancy in a bi-lingual collection of news articles by clustering them based on semantic similarity using an incremental nearest neighbor algorithm. The facts extracted from English language articles are bootstrapped within each cluster to extract the facts from comparable Hindi language articles using the Google Translator API. This bootstrapping method within the cluster aids in identifying related sentences containing new information from a low-resource language like Hindi. Using this approach, information from news articles in any low-resource language can be extracted without the use of language-specific tools such as Parts-Of-Speech (POS) taggers, Named Entity Recognizers, and Open Relation Extractors, making it more suitable for resource-deficient Indian languages. Experiment results show that the proposed framework extracts new facts from Hindi news articles with a high recall rate.

Finally, the proposed method employs a multi-layer perceptron-based classifier to determine whether or not a given triple is genuine. It attempts to vectorize the triples by employing both frequency and probability-based word embedding models. These two embedding models help in considering both word and document level features while reducing vector dimensionality. Empirical results show that the proposed classifier outperforms the baseline classifiers in prediction accuracy.

Keywords: Information Extraction, Knowledge base construction, Knowledge base completion, Fake news classification, Machine learning, Triples, Bootstrap, Cluster, Natural language processing, Knowledge representation.

CONTENTS

List of Figures viii

List of Tables ix

List of Algorithms xi

List of Abbreviations xiii

1 Introduction 1

1.1 Applications 6

1.2 Motivation 7

1.3 Thesis Contributions 10

1.4 Organization of the Thesis 11

2 Literature Review 13

2.1 Knowledge, Knowledge Base, and Knowledge Base Management . . . 13

2.2 Knowledge Base Construction 14

2.2.1 Knowledge Extraction 15

2.2.2 Knowledge Representation 20

2.3 Integration 21

2.4 Validation 22

2.5 Research Challenges 24

2.6 Summary of knowledge base construction systems 27

3 Problem Description 31

3.1 Problem Statement 31

3.2 Objectives 31

3.3 System Architecture 34

4 Construction of a Bootstrapping knowledge base 35

4.1 Introduction 35

4.2 System Architecture . . . 37
4.3 Data Acquisition . . . 37
4.4 Information Extraction . . . 40
4.4.1 Rule based Approach . . . 40
4.4.2 DBpedia based Approach . . . 43
4.5 Semantic Merging . . . 47
4.5.1 Similarity between text entities . . . 47
4.5.2 Similarity between Image entities . . . 50
4.6 Knowledge Base Population . . . 50
4.7 Experimental Analysis . . . 54
4.7.1 Evaluation of topic modeling and knowledge base aided data gathering . . . 55
4.7.2 Evaluation of Information Extraction . . . 56
4.7.3 Evaluation of Semantic Merging . . . 57
4.8 Summary . . . 58
5 A Knowledge Base Completion Framework **61**
5.1 Introduction . . . 61
5.2 System Architecture . . . 62
5.3 Clustering . . . 62
5.4 Extraction . . . 66
5.4.1 Candidate sentence identification . . . 66
5.4.2 New triple generation . . . 66
5.5 Experimental Results . . . 67
5.5.1 Evaluation of Clustering Algorithm . . . 67
5.5.1.1 Bi-lingual Evaluation . . . 69
5.5.1.2 Mono-lingual Evaluation . . . 70
5.5.2 Evaluation of Extraction . . . 71
5.6 Summary . . . 74
6 Triple based fake news classification **77**
6.1 Introduction . . . 77

6.2 System Architecture . . . 79

6.3 Model construction . . . 79

6.3.1 Data modeling . . . 80

6.3.2 Investigation of baseline model . . . 82

6.3.3 Implementation of Multi-layer Perceptron Model . . . 84

6.4 Experimental description . . . 86

6.4.1 Dataset and Evaluation metrics . . . 86

6.4.2 Implementation environment . . . 87

6.4.3 Evaluation of model performance . . . 88

6.5 Summary . . . 92

7 Conclusions and Future Scope **93**

Bibliography **98**

Publications **113**

LIST OF FIGURES

1.1 An example knowledge graph 2

1.2 Knowledge base creation pipeline 3

2.1 Illustration of KB and KG 14

2.2 Taxonomy of Knowledge base management 14

2.3 Taxonomy of methods for knowledge base facts validation 23

3.1 Overview of the proposed model to create a crime knowledge base . . . 34

4.1 Overall Architecture 37

4.2 Topic Modeling and Knowledge base aided data acquisition 39

4.3 Rule based Generation of Entities and Relations 40

4.4 NER tagged tuples generated by the rule based approach 43

4.5 Illustration of triple extraction with an example using DBpedia based Approach 45

4.6 *Semantic merging of events and entities* 48

4.7 *Ontology structure* 53

4.8 *Entity graph without image* 53

4.9 *Entity graph with image* 54

4.10 Redundant entities before and after semantic merging 59

5.1 *Detailed architecture of the proposed framework* 63

5.2 *Candidate sentence generation* 67

5.3 *New triple generation* 68

5.4 Bi-lingual evaluation of proposed clustering Algorithm: Silhouette coefficient for varying number of events 68

5.5 Precision for five clusters 72

5.6 Recall for five clusters 74

6.1 Overview of the proposed work 80

6.2 Proposed data modeling approach 82

6.3 Multi-layer perceptron model 86

LIST OF TABLES

2.1 Open source NLP tools 17

2.2 Events and their description 25

2.3 Similarity scores between the events 25

2.4 Features supported by mono-lingual systems 29

2.5 Features supported by multi-lingual Systems 30

4.1 Rules for NER tagging 44

4.2 Data gathering results for a sample of six articles 56

4.3 Comparison of NLTK and proposed DBpedia-based approach 57

4.4 Knowledge base before and after semantic merging- using only semantic similarity measure 58

4.5 Knowledge base before and after semantic merging- using both contextual and semantic similarity measure 58

5.1 Number of related events for keywords before and after clustering . . . 70

5.2 Bi-lingual evaluation of proposed clustering algorithm: Time taken for clustering 70

5.3 Mono-lingual evaluation of proposed clustering algorithm 71

5.4 Comparison of MTML_KBC and proposed approach 73

6.1 Confusion matrix for Multinomial NB classifier with TF-IDF 87

6.2 Comparison of classifiers without named entity tags 89

6.3 Comparison of classifiers with named entity tags 90

6.4 Accuracy values for 10 test samples for the base line model and MLP model 91

LIST OF ALGORITHMS

1 Rule based Algorithm for Entity and Relation extraction 42

2 DBpedia based Algorithm for Entity and Relation extraction 46

3 Algorithm for Knowledge Base Population 52

4 STNN Algorithm 65

LIST OF ABBREVIATIONS

Abbreviations	Expansion
IE	Information Extraction
KB	Knowledge Base
BKB	Bootstrapping Knowledge Base
KBC	Knowledge Base Construction
NLP	Natural Language Processing
NER	Named Entity Recognition
NET	Named Entity Tags
RE	Relation Extraction
KG	Knowledge Graph
KBM	Knowledge Base Management
LEA	Law Enforcement Agencies
CKB	Crime Knowledge Base
CIA	Crime Intelligence Analysis
BIA	Business Intelligence Analysis
UE-tagged	Un-tagged and Erroneously tagged
MT	Machine Translation
RDF	Resource Description Framework
RDFS	RDF Schema
SPARQL	Sparql Protocol and RDF Query Language
LOD	Linked Open Data
SW	Semantic Web
NLU	Natural Language Understanding
KE	Knowledge Extraction
NLTK	Natural Language Tool Kit
OIE	Open Information Extraction
ORE	Open Relation Extraction
KR	Knowledge Representaion
PoS	Parts of Speech
URI	Uniform Resource Identifier
OWL	Web Ontology Language
DL	Description Logic
QL	Query Languages

Abbreviations	**Expansion**
RDQL	RDF Data Query Language
SeRQL	Second Generation RDF Query Language
GLOO	Graphical query Language for OWL Ontologies
OntoVQL	Visual Query Language for OWL Ontologies
SAIQL	Schema And Instance Query Language
MPI	Message Passing Interface
MNB	Multinomial Naive Base
PA	Passive-Aggressive
SVM	Support Vector Machine
KNN	K-Nearest Neighbors
LR	Logistic Regression
RF	Random Forest
MP	Multi-layer Perceptron
ANN	Artificial Neural Networks
RM	Relevance Measure
CRF	Conditional Random Fields
HMM	Hidden Markov Models
MEMM	Maximum Entropy Markov Model
API	Application Programming Interface
BERT	Bidirectional Encoder Representations from Transformers
ANNIE	A Nearly-New Information Extraction System
ML	Machine Learning
LDA	Latent Dirichlet Allocation
LCS	Least Common Subsumer

CHAPTER 1

INTRODUCTION

Due to the emergence of the internet, a vast volume of unstructured data is generated and published digitally daily like online newspapers. However, such unstructured data which is available in human-readable format is unsuitable to be read and processed by the machine automatically for applications like question answering. Berners-Lee who introduced the concept of SW defines the vision of SW as - "Enrich human-readable web data with machine-readable annotations, allowing the Webs evolution as the biggest database in the world" (Kuck (2004)). This triggered the researchers in the semantic web community to create a structured form of data from unstructured sources in machine-readable form by emphasizing the semantics of the data which provides global data integration in the World Wide Web. This gave birth to many of the open or cross-domain KBs like DBpedia (Lehmann et al. (2015)) and domain-specific KBs like Music Knowledge Base (MKB) (Oramas et al. (2016)).

The goal of Knowledge Base Construction (KBC) is in extracting knowledge from unstructured texts and storing it into a KB as an instance of predefined taxonomy or ontological structure. KB is a special kind of graph database that stores machine-readable knowledge in the form of graphs called Knowledge Graphs (KGs). KG is a set of triples knows as Resource Description Framework (RDF) facts (Candan et al. (2001)) where each triple in the form of $< head - relation - tail >$ or $< subject - predicate - object >$ represents a relationship r between a head entity h and a tail entity t. An

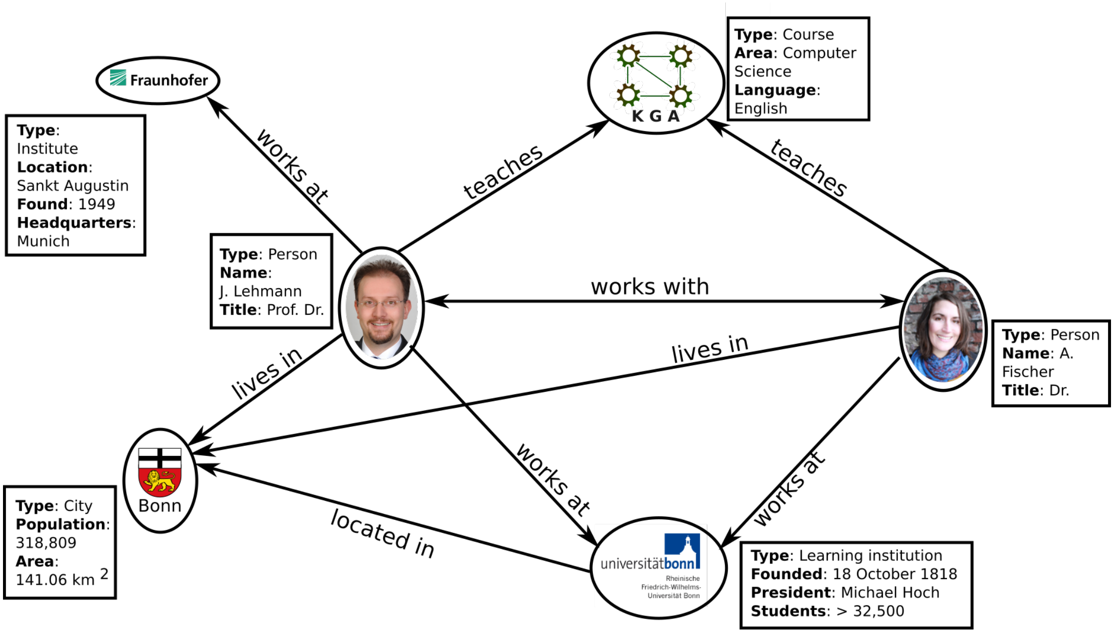


Figure 1.1: An example knowledge graph

example of KG with entities and their relationships is shown in figure 1.1[1]. Generation of graphs requires transforming the syntactic structure of natural language texts to their semantic equivalents by using Natural Language Processing (NLP) techniques like Named Entity Recognition (NER) and Relation Extraction (RE) (Martinez-Rodriguez et al. (2018)).

Currently, crime monitoring and prevention is of great interest to most LEAs across the globe to keep the world safe. Although valuable, authentic, and timely information in online newspapers can be extracted and grouped manually by reading through all the available newspapers, this is a difficult and error-prone job that needs a lot of human resources. Moreover, a single resource representing the information in multiple documents can provide significantly more semantic information than is available from the documents considered independently. Even though LEAs have information available with them that comes within their jurisdiction, prevention or tracking of the criminal activities is limited to some specific regions. With access to a shared KB having information collected from various sources, helps them to prevent or track criminal activities in other regions also. Hence a central repository of entities to access complete details

[1] https://github.com/SmartDataAnalytics/Knowledge-Graph-Analysis-Programming-Exercises

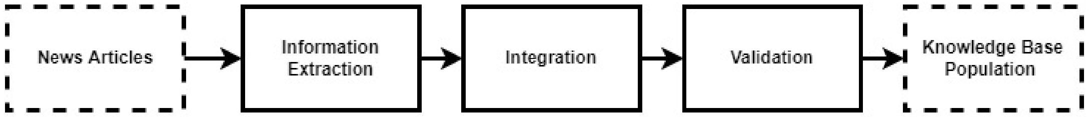


Figure 1.2: Knowledge base creation pipeline

about an entity is in high need. In this work a bootstrap-based model for developing Crime Base, a Crime Knowledge Base (CKB), a repository of crime-related facts i.e. entities and their relationships extracted from diverse news articles is proposed.

A primary challenge in developing a model to automatically generate a CKB is in enriching the KB with text as-well-as image data extracted and integrated from multiple online news articles without any loss and redundancy, unlike the existing works that consider only text information(Alruily et al. (2014); Arulanandam et al. (2014); Chau et al. (2002); Dasgupta et al. (2017)). The KB so enriched, provides complete information about an entity in a single place without language and modality barriers. The general pipeline followed for the generation of a KB is shown in figure 1.2. The three fundamental steps in any KB generation are Information Extraction (IE), Integration, and Validation, and they continue to receive a great deal of attention in the domain of KBC (Buitelaar et al. (2008)).

IE is the process of identifying entities and their relationships from multiple sources using NLP techniques such as NER and RE (Martinez-Rodriguez et al. (2018)). NER is the task of identifying and categorizing real-world entities like the name of a person in a text. It is also known as entity chunking or extraction. Each detected entity is assigned to one of several domain-specific or domain-independent categories also known as named entity tags such as PERSON, LOCATION, ORGANIZATION, and so on. For example, a NER model might recognize the word “Bill Gates” in a text and classify it as a “PERSON”. Whereas, the task of extracting semantic relationships from a text is known as RE. Extracted relationships are typically formed between entities of fixed types and fall into one of several semantic categories, such as $Works\_For$, which represents a relationship between entities of the types PERSON and COMPANY. Many tools and techniques are available to perform named entity recognition (Goyal et al. (2018)). These tools and methods, on the other hand, have not investigated the effect of Information Extraction (IE) on integration. As a result, any Un-tagged and Erroneously tagged (UE-tagged) entities generated during the extraction process, introduce redundancy and loss of information in the final knowledge base after integration. Untagged entities are those that do not have tags assigned to them. Erroneously tagged entities, on the other hand, have been assigned different tags in different sentences. For instance, a well known NLP tool, such as the Natural Language Tool Kit (NLTK) assigns a LOCATION tag to “Delhi” for the phrase “Delhi Police”, without assigning any tag to “Police”. Whereas the phrase “Lawrence Bishnoi Gang member” tagged by NLTK as either ORGANIZATION or PERSON based on the syntactic variation in the sentence in which the phrase is used. Un-tagged entities will be ignored from adding to the knowledge base while integrating information from multiple sources, resulting in information loss. Similarly, erroneously tagged entities will be added to the knowledge base multiple times as a result of incorrect tags assigned to them in different sentences, resulting in redundancy. In the literature, erroneously tagged entities are also referred to as wrongly or incorrectly tagged entities.

The enrichment of information from multiple sources to provide complete details of an entity in one place by removing duplicates and identifying new information related to an entity is referred to as information integration (Dragos (2013)). Identifying the semantic correlation between different entities is a significant challenge in the integration of information from multiple sources. This is accomplished by using either corpus or knowledge base methods by the existing works. However, utilizing the power of both the similarity measures is not explored by the extant works (Zhu and Iglesias (2018)). For instance, the phrases “death at the bank of a river” and “fraud at canara bank” are semantically non-related and hence are also dissimilar. Similarly, the phrases “ATM fraud at canara bank” and “robbery at SBI bank” are semantically related or contextually similar but are not semantically similar. Whereas, the phrases “ATM fraud at canara bank” and “ATM robbery at canara bank” are contextually similar as well as semantically similar. Contextual similarity measure helps to identify the semantic relatedness between the events based on the context in which the entities participating in the events are used. To ensure the need for both the similarity measures, an experiment is conducted over several sentences. Empirical evidence suggests that corpus-based contextual similarity models such as Word2Vec can better capture the contextual meaning of entities. However, these models do not provide a higher similarity score for semantically similar events. In contrast, knowledge base algorithms such as the Wu and Palmer method provide a higher similarity score for semantically similar events that are contextually similar. As a result, to capture more semantics about the entities, it is necessary to consider the contextual meaning of the entities as well as their synonymic relations. Hence, this work investigates the utility of contextual and semantic similarity measures in determining entity similarity.

In the era of a multi-lingual environment(Akbik et al. (2016)) where the information is scattered across the web in multiple languages, most of the facts are redundant but are enriched with some new facts. For instance, the online news articles from different sources with various native languages within the same window of published dates, include information related to almost similar facts. However, each source may be enriched with some new facts about an event, and failing in identifying such facts is censorious for applications like crime prevention and monitoring. Hence, for applications that develop KB from news articles, it is not sufficient to extract the information only from English news articles to ensure the completeness of KB.

Finally, it is also necessary to validate the facts as fake or true. Even though checking for the trustworthiness of the knowledge base facts are studied by some of the recent works like (Jia et al. (2019)), here the scope is limited to check the semantic relatedness of facts. For instance, these models determine “whether the relationship r occurs between the entity pair $< h, t >$” and such facts are inferred from the existing facts in the knowledge graph. These models, however, cannot detect any fake facts added to the knowledge base. Hence, it is also necessary to check the truth-ness or falsity of facts before they are added to the knowledge base. A general approach to verify whether the given news is fake or not would be checking from the origin itself. The problem of tracing the origin is NP-Hard. Hence, the recent works to detect fake news are based on developing machine learning models over the texts. As these models are trained considering the original sentences in the raw text, they are not applicable to classify a given triple as true or fake which is most necessary to enrich the knowledge base with trustworthy facts. Hence in this work, the effectiveness of using triples for fake news detection using six machine learning and one deep learning-based classifiers is empirically studied.

It is also critical to populating the knowledge base with an enormous amount of integrated information using a knowledge representation model. Among many knowledge representation models like distributed, symbolic, probabilistic, and rule-based, ontology is now widely used in many areas like artificial intelligence, biomedical informatics, semantic web, system engineering, forensic data analysis, and information architecture (Jalil et al. (2017)). Knowledge base represented in the form of ontology can be used further by many of the data mining tasks such as clustering. Clustering entities based on ontological relationships like hypernym (Dhuria et al. (2016)), only part of the ontology can be accessed by the users from the whole knowledge base.

1.1 APPLICATIONS

Here we present some of the applications where the KB plays an important role:

- **Question Answering:** The task of question answering involves answering a natural language question posed by the users using the information stored in KBs (Calijorne Soares and Parreiras (2020); Moschitti et al. (2017)). The difficulty of question answering systems is in understanding the natural language queries written in users native languages. Hence, the natural language queries are converted to formal queries using query languages such as Sparql Protocol and RDF Query Language (SPARQL) (Hazrina et al. (2017)) over a global schema described by the ontology (Óscar Ferrández et al. (2009)). IE techniques are also applied to extract the facts present in queries and the same or related facts are searched in KBs to retrieve the answers suitable for the query. "Watson" is a popular question answering system developed in IBM's DeepQA project [2].
- **Information Retrieval:** Keywords are used to index and retrieve the documents by the traditional Information Retrieval (IR) systems (Otegi et al. (2015)). Vo-

[2] https://www.ibm.com/in-en/watson

cabulary mismatch problems incurred using the keywords are overcome by using the knowledge-based semantic relatedness techniques. Here the concepts and instances present in KBs act as indexes for documents. Instead of searching for all documents, the concepts and instances related to the given query are searched in KBs and only those documents are retrieved for a user query. This reduces the time of retrieval for a given query and also minimizes the vocabulary mismatch due to the high semantics hidden in KBs.

- **Text Summarization:** It is a long-standing research problem in the field of Natural Language Understanding (NLU) to generate a precise summary of a given text. Here the KBs are employed to identify the important concepts to be included in the summary that expresses the meaning of the whole document. In (Timofeyev and Choi (2018)) a system for summarization is developed using the Cyc development platform which consists of the world's largest KB and powerful inference engines. The system produces summarization for a given document using knowledge acquisition, discovery, and representation. (Rashidghalam et al. (2016)) uses BabelNet KB is used to generate the summaries.

- **Search**: KBs are used in many e-commerce applications to assist users search for a specific product. For example, a user may be interested in searching for a Laptop. Here KBs are employed to search for all related products like a wireless mouse by searching for the categories related to Laptop. Google introduced the term knowledge graph and used it as a means to improve its search results. Companies like Facebook and Microsoft have their knowledge graph which is non-public (Heist et al. (2020)). Some works makes use of publicly available knowledge graphs like Wikipedia to develop their search interfaces (Milne et al. (2007)).

1.2 MOTIVATION

In the current era of the internet, information related to crime is scattered across many sources namely news media, social networks, blogs, video repositories, etc. Crime reports published in online newspapers are often considered reliable compared to crowdsourced data like social media and contain crime information not only in the form of unstructured text but also in the form of images. Furthermore, information in online newspapers is available not only in English but also in other native languages. Given the volume and availability of crime-related information present in online newspapers, extracting and integrating crime entities from multiple modalities and languages, and representing them as a knowledge base in the machine-readable form will be useful for any law enforcement agency to analyze and prevent criminal activities.

The extraction and integration of text and image data from multiple online news articles without loss and redundancy is a major challenge in developing Crime Knowledge Base, in contrast to existing works that consider only text information, such as (Dasgupta et al. (2017)). Furthermore, the existing works to generate the knowledge base consider extraction and integration of information as two independent problems (Buitelaar et al. (2008)). As a result, techniques and tools for NER (Goyal et al. (2018)) used by the works ignored the impact of Un-tagged and Erroneously tagged (UE-tagged) entities generated by the IE over the integration. Due to this, loss and redundancy of information incurred in the final KB are unexplored. For example, an entity that is not tagged will not be added to the KB during the integration process, resulting in information loss. Similarly, an entity assigned multiple tags will be added to the KB more than once, resulting in redundancy after integration. From the standpoint of knowledge base construction, it is critical to minimize UE-tagged entities to reduce redundancy and information loss in the final knowledge base. Even though the existing Relation Extraction (RE) tools and techniques (Augenstein et al. (2016); Zouaq et al. (2017)) extract the triples of high accuracy, they do not assign any tags to the entities. As a result, joint modeling of the NER and RE to extract entities and their relations is crucial.

Most works on knowledge base creation ignored the problem of integration to ensure non-redundancy in the KB by identifying a set of semantically similar entities (Wu et al. (2015)). The works that addressed the issue of integration, such as (Buitelaar et al. (2008)), used either contextual or semantic similarity measures alone to identify the similarity between the entities of two events while integrating the facts from multiple sources. As a result, complete semantics hidden in facts is ignored. To ensure the need for both the similarity measures, empirically it is found that the contextual meaning of the entities can be better captured using corpus-based contextual similarity models like Word2Vec (Church (2017)). However, these models do not provide a higher similarity score for semantically similar events. In contrast, the knowledge base algorithms like Wu & Palmer method (Wu and Palmer (1994)) provide a higher similarity score for semantically similar events provided they are contextually similar. Hence, it is necessary to consider the contextual meaning of the entities along with their synonymic relations to capture the full semantics of the events.

For the news articles scattered across the web in multiple languages, it is also necessary to ensure the completeness of the knowledge base by incrementally extracting the new facts from articles other than the English language. To date, the knowledge base completion is studied as a problem of knowledge refinement where the missing facts are inferred by reasoning about the information already present in the knowledge base (Pezeshkpour et al. (2020)). However, facts missed while extracting the information from multi-lingual sources are ignored. Existing works such as (Gerber and Ngomo (2012)) accomplished this by utilizing language processing tools such as Parts of Speech taggers or translating articles from other languages into English. However, this is not feasible for resource-limited Indian languages where language processing tools are either unavailable or inaccurate. Furthermore, it is not feasible to translate the entire article into English. As a result, a method for knowledge base completion that enriches the KB with new facts from low-resource language articles without the use of language processing tools is critical.

From a knowledge base construction perspective, it is also most essential to check the credibility of facts before they are added into the knowledge base. Existing works for determining the validity of facts are limited to evaluating the semantic correctness of facts (Jia et al. (2019)). To date, there is no model to classify a given knowledge base fact or triple as false or true. In recent years, the subject of detecting fake news has been explored as a classification problem, where a news piece is classed as either true or false. Current efforts to detect fake news rely on the development of sophisticated machine learning and deep learning models based on a variety of news datasets (Agarwal et al.

(2020)). These models, on the other hand, are not intended to train knowledge base facts and thus are unsuitable for classifying a given triple as true or false. As a result, it is critical for a model that generates a news KB to implement a machine or deep learning model to classify a triple as true or false so that the KB is enriched with validated facts, which is critical for applications such as crime monitoring.

1.3 THESIS CONTRIBUTIONS

1. An algorithm for extraction of crime entities and their relations from English language news articles is proposed. Here the extraction is focused on minimization of un-tagged and incorrectly tagged entities and hence reduction of redundancy and loss of information in the final knowledge base. This is achieved in the current work by using DBpedia as a background KB.

2. The imaging modality data is explored in constructing the KB. The image captions are used to extract the image entities from images and both image captions, as well as low-level features of images, are considered for integrating the images from multiple news articles.

3. Use of both contextual as well as semantic similarity measures for semantic integration of crime-related text and image entities are investigated.

4. A generic framework for KB completion is proposed to enrich the KB with the new facts extracted from related news articles in the Indian language Hindi. As a primary stage of the framework, a clustering algorithm is proposed to group a set of related news articles from multiple languages. As a second stage, a bootstrapping approach is proposed that extracts the new facts from low-resourced language articles within each cluster. Using the framework, a parallel corpus of multi-lingual news articles can be developed which can be used for many NLP tasks like Machine Translation (MT). Similarly, entities from resource deficit language news articles can be extracted without using the respective language processing tools and using an appropriate MT tool.

5. Fake news classification problem is studied as a classification of KB facts or

triples as fake or true. Here, a first attempt is made to validate the knowledge base facts using a Multi-layer Perceptron with state-of-the-art Word2Vec (Church (2017)) and GloVe (Pennington et al. (2014)) word embeddings along with traditional TF-IDF and count vectorizers. Investigated a machine learning-based baseline model for comparison by implementing six machine learning-based algorithms trained over the triples. The significance of named entity tags in validating the facts is also investigated by training the models using triples with and without named entity tags.

1.4 ORGANIZATION OF THE THESIS

The rest of the thesis is organized as follows. Chapter 2 provides a detailed survey of the existing research works in the domain of knowledge base construction and representation. In addition, the chapter provides a list of research gaps identified from the existing works. Chapter 3 describes the research problem that is the focus of this thesis. Chapter 4 describes how an initial version of the KB is constructed from English language news articles and describes our proposed approach for extracting and integrating the text and image entities. The chapter also gives the experimental analysis of our approach in terms of accuracy and KB size. Chapter 5 discusses the proposed clustering and bootstrapping-based KB completion framework to enrich the KB with the new facts available from Hindi-language news articles. The chapter discusses the clustering algorithm for grouping the related articles from a bi-lingual corpus and bootstrapping approach for extracting the facts from low-resourced language. An experimental evaluation of the proposed framework is also presented. Chapter 6 discusses how fake news classification is addressed using the KB facts or triples. The chapter describes the use of one deep learning-based Multi-layer perceptron and six machine learning-based models in classifying a given triple as fake or true. Experimental results regarding the triple classification with and without named entity tags are also presented. Finally, Chapter 7 presents a summary of the research work presented in this thesis and suggests some future research directions.

CHAPTER 2

LITERATURE REVIEW

This chapter covers a structured review of the various state-of-the-art techniques and related research works for knowledge base construction, Integration, and validation. The section 2.1 briefly defines the knowledge, knowledge base, and knowledge base management. The techniques and tools for knowledge base construction are discussed in detail in section 2.2. The section 2.3 describes the methods for knowledge base integration. Knowledge base validation and its need from a knowledge base construction perspective are explained in section 2.4. The chapter also consolidates existing knowledge base generation systems based on the features supported by each, highlighting the research gaps. Finally, the chapter discusses the research challenges discovered during the literature review.

2.1 KNOWLEDGE, KNOWLEDGE BASE, AND KNOWLEDGE BASE MANAGEMENT

Knowledge is a collection of real-world entities and their relations stored as a KB. Nowadays Knowledge Graphs (KGs) have drawn great attention in the domain of Knowledge Base Management (KBM). The terms KG and KB are used interchangeably in the existing literature and are different only in the way the knowledge is represented (Ji et al. (2021)). Figure 2.1 illustrates the representation of knowledge using KG and KB with a simple example. KB represents the knowledge in terms of triples like $< e_1 - R - e_2 >$. KGs are the graphical representations of the knowledge where the nodes represent the entities and the relations are represented using edges. Like Database management,

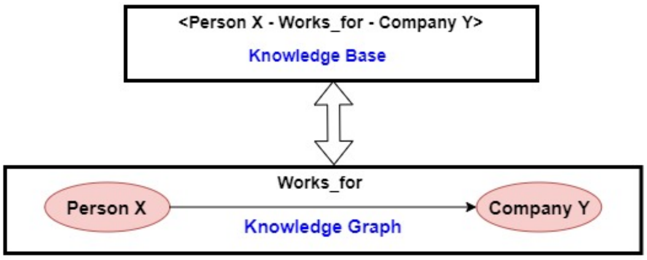


Figure 2.1: Illustration of KB and KG

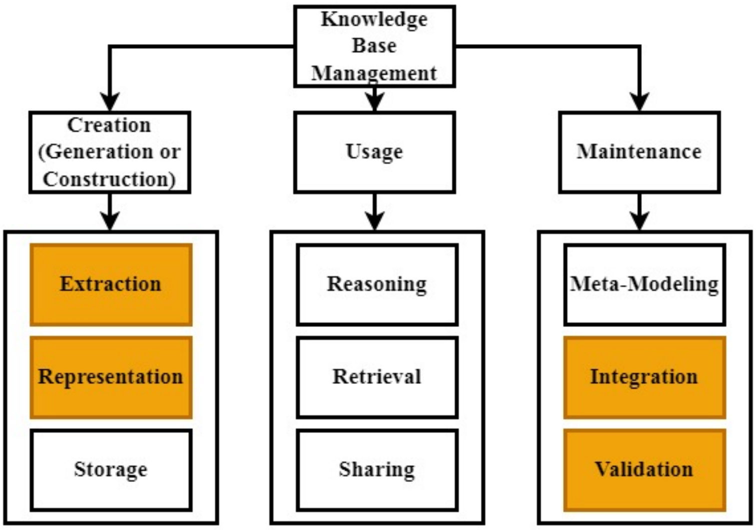


Figure 2.2: Taxonomy of Knowledge base management

KBM involves the creation, retrieval, and maintenance of KB. Based on the existing survey (Martinez-Gil (2015)), the KBM is summarized as shown in the figure 2.2. The scope of the thesis is limited to Knowledge Base Construction (KBC) mainly involves KB creation and maintenance and is darkened in the figure 2.2. The detailed survey of the existing tools and techniques for KBC are considered in the following sections.

2.2 KNOWLEDGE BASE CONSTRUCTION

Knowledge Base Construction (KBC) is the main vision of SW to create a shared repository of KB in machine-readable form. The existing works for KBC follow either manual or automatic creation of KBs. The degree of ambiguity is very less in manually constructed KBs due to the involvement of domain expert knowledge engineers and are suitable for a particular domain. However, it is difficult to extract more facts with only manual efforts and hence are not preferred in many applications. Best known manually created KBs include lexical resources like WordNet (Miller (1995)), FrameNet (Baker et al. (1998)) and VerbNet (Schuler (2005)). WordNet groups the words into synsets and provides fixed relations among them like hypernymy. FrameNet describes the frame structure for words. Each frame represents a verb term or predicate with a list of frame elements that represent the semantic arguments of the predicate. VerbNet maps verbs to their corresponding Levin classes. To name a few other manually created KBs to include MusicBrainz and Discogs (Oramas et al. (2016)) in the music domain. Many of the works create the KB automatically without human intervention and are the main interest of study in the domain of KBC. Knowledge Extraction and Representation are the two fundamental steps in any KBC and are surveyed in detail in the following sections.

2.2.1 Knowledge Extraction

Knowledge Extraction is a method of extracting information using well-known tools and techniques for information extraction.

Information Extraction techniques concentrate on extracting some specific and predefined entities and their relationship between them called NER and RE respectively (Derczynski et al. (2015))(Martinez-Rodriguez et al. (2018)). Data mining and crowdsourcing are the Standard techniques used by several researchers in the area of crime information extraction. Entity Extraction, Clustering Techniques, Association Rule, Classification Techniques, and Social Network Analysis are the typical data mining techniques used for crime information extraction and analysis (Hossein et al.). These techniques are used to identify the most complex crime patterns along with identifying the crime-related entities. Here, the discussion is limited to entity extraction techniques as the extraction of entities is the basic requirement for integrating the information from multiple newspapers. Authors in (Furtado et al. (2010)) used crowdsourcing where they provide a platform for individual users to report crime-related information and is available to all other users to view and comment. The main problem in such an environment is difficulty in verifying the reliability of reported crimes. The solution provided by authors to address this issue does not apply to data generated by online newspapers and applies to only the platform they created.

The existing approaches to achieve NER are based on hand-crafted Lexicons and Rules, Statistical models, and Machine learning algorithms (Chau et al. (2002)). Lexical lookup-based approaches maintain a list of hand-written lexicons which contain some known entities of interest. These systems check for lexicons in the given sentence that match any of the lexicons in the list to identify the entities in the given sentence. Rule-based approaches use hand-crafted rules to identify the entities. Systems based on Statistical methods make use of some training data along with statistical models like Conditional Random Fields (CRF) to identify specific patterns for entities in texts. Machine learning-based systems use some machine learning algorithms like Hidden Markov Models (HMM), Maximum Entropy Markov Model (MEMM), Neural Networks, and Decision Trees to identify and extract patterns from texts. Statistical models and machine learning algorithms need a large set of training data to achieve accuracy. Whereas, developing the lexicons of all categories in the crime domain to cover the entities extracted from various sources is an impossible task. There has been much work in adopting NER for identifying entities that belong to the crime domain. For example, in (Arulanandam et al. (2014)), authors used NER to identify locations and adopted CRFs to assign each sentence in an article either as "crime location sentence" or "not a crime location sentence". Many of the works like (Ku et al. (2008)) used lexical lookup-based approaches along with NLP Techniques like POS tagging for extracting crime-related data from various sources. But these works extract the entities without focusing on the impact of the extracted entities on the integration of information from multiple sources.

There are some well-known off-the-shelf open-source NLP tools like NLTK to perform NER and are used by many of the knowledge base construction systems such as (Dasgupta et al. (2017)) to extract the information. The table 2.1 consolidates the various tools and their brief description for each.

The most important step in IE is RE and is a difficult task as the relations are not expressed uniformly in natural language texts. For example, phrases- "X born

[1] https://spacy.io/
[2] https://opennlp.apache.org
[3] https://allennlp.org
[4] https://nlp.johnsnowlabs.com

Table 2.1: Open source NLP tools

Name	Description
Natural Language Toolkit (NLTK) (Loper and Bird (2002))	It is a well-known tool used by many of the researchers in industry and academia to performs almost all the NLP tasks like Tokenization, POS tagging, Stemming, Named Entity Classification, Parsing and so on in Python language. It provides libraries trained over more than 50 corpora.
Spacy [1]	This is used to work with real-world products and also developed as a library for python language.
Apache Open NLP [2]	Along with basic NLP tasks, it also performs more advanced NLP tasks such as language identification, perceptron based machine learning and so on.
Core NLP (Manning et al. (2014))	It is a Java language based open source tool developed by Stanford NLP group for basic NLP tasks.
Allen NLP [3]	AllenNLP is developed on PyTorch and can be used for building machine learning models for core NLP tasks like textual entailment.

Flair (Akbik et al. (2019))	It is a deep learning based framework that provides an easy to use Application Programming Interface (API) and also includes word embedding models like GloVe.
SparkNLP[4]	It is a NLP library built over Apache Spark ML and supports BERT like transformers.
General Architecture for Text Engineering (GATE) (Cunningham et al. (1997))	It includes ANNIE, an IE system to perform basic IE tasks.

in Surathkal" and "X birthplace is Surathkal", both represent the BornIn relationship which is difficult to extract due to natural language variations. In this regard, most of the works for RE came up especially in Open IE (OIE) (Banko et al. (2007)) category to extract relations from an open domain. To name a few powerful OIE systems, TEXTRUNNER (Banko et al. (2007)), PATTY (Nakashole et al. (2013)), REVERB (Fader et al. (2011)), PROSPERA (Nakashole et al. (2011)), and KNOWITALL (Etzioni et al. (2004)) extract the relations of high accuracy from open web in English language.

The existing approaches for RE use any of the methods from rule-based, supervised, un-supervised, semi-supervised, distant supervised, and bootstrap. Few of the approaches use some combination of these methods. Rule-based methods take the advantage of the syntax defined in the texts and use hand-crafted rules to extract the relations (Fader et al. (2011)). This method can be applied to extract relations from a specific domain as the set of relations and their patterns are known in advance which is not possible for open-domain texts. Even though supervised methods extract complex relations, they need a lot of pre-defined labeled texts as training data to generate relation extractors that extract relations from un-labeled texts (Zelenko et al. (2003)). To reduce the human efforts in creating the labeled data semi-supervised methods used a few sets of labeled data as a seed to train the model. However, the accuracy of the methods depends on the seeded data selected for training (Rozenfeld and Feldman (2008)). In this move, distant supervision methods avoid the manual creation of labeled data by using facts from background KBs like DBpedia as the labeled data to obtain the training samples with an assumption that- “if two entities participate in a relationship in the background KB, any sentence that contains those two entities might express that relation”. However, the success of the method depends on how good the assumption holds (Augenstein et al. (2016)). Similar to rule-based methods, un-supervised and bootstrapping methods do not rely on creating the training data. Un-supervised methods extract words between entities, cluster them and produce relation-strings (Banko et al. (2007)). Bootstrapping methods use a background KB to extract predicate-based patterns that involve entities present in the background KB (Gerber and Ngomo (2012)).

Apart from the individual efforts in generating KBs like (Oramas et al. (2016)), a few integrative projects involve a community of users in creating KBs by extracting and updating the facts from crowd-sourced data like Wikipedia has also emerged. To name a few, Yago (Rebele et al. (2016)) is a KB created automatically from Wikipedia, WordNet, and Geonames. DBpedia (Lehmann et al. (2015)) exploits both free text as well as semi-structured data like infoboxes from Wikipedia to create the KB. BabelNet (Navigli and Ponzetto (2012)), the largest repository of multi-lingual words and senses, integrates Wikipedia and WordNet for creating the KB. Wikidata (Erxleben et al. (2014)) is a KB enriched with facts extracted only from Wikipedia. Even though the KB generated by these systems is well structured to support a web of linked data, the facts covered and validated by these systems are limited to Wikipedia.

There are some open-source tools like FRED (Gangemi et al. (2017)) that generate structured knowledge graphs from unstructured texts. Comparison of a set of knowledge extraction tools is performed in (Gangemi (2013)). Among all the tools, FRED is a powerful tool that extracts knowledge from 48 languages. However, the capability of the tool is limited to only extraction and lacks in integrating the knowledge extracted from multiple languages.

2.2.2 Knowledge Representation

A common representation for web data and its resources is a fundamental requirement in SW which allows sharing knowledge across applications and community boundaries. Ontologies play an important role in this direction (Poli et al.). The idea behind ontologies is to explicitly define the knowledge by providing commonly used concepts and their relationships in a particular domain. KBs are created as an instance of explicit Knowledge defined by the ontologies. Knowledge Representation (KR) is the study of how knowledge about the world can be represented, and what kinds of reasoning can be done with that knowledge i.e. it provides a data model and languages to create, store and access the KBs that are machine-readable and accessible across the applications. An overview of the data model and languages is briefly explained as follows.

- **RDF and RDFS:** RDF is a standard data model (Candan et al. (2001)) for interchanging data on the Web. It defines a simple triple-based graphical data model like $< subject{-}predicate{-}object >$ to represent the resources on the web. RDF provides Uniform Resource Identifiers (URIs) to identify the resources. RDF Schema (RDFS) is an RDF vocabulary description language, built on top of RDF, is a general-purpose language for representing information on the web. However, RDFS is rather simple and it still does not provide the exact semantics of a domain.

- **OWL:** The Web Ontology Language (OWL), is an ontology language with formally defined meaning for the SW (McGuinness et al. (2004)). OWL lays on top of RDF and RDFS and provides stronger syntax and rich vocabulary. All of them have similar foundations, but the machine interpretability of OWL is stronger than RDF. Like RDFS, it can be used to define classes and properties, but an additional set of constructs provided by OWL increases its expressive power. OWL is a Description Logic (DL)-based ontology language and hence provides more inferencing capability with the help of DL-reasoners.

- **Query Languages:** Query Languages (QL) are used to request data from data repositories. QLs in SW can be divided into RDF-based QLs and OWL DL-based

QLs. A few examples of RDF-based QLs include RDF Data Query Language (RDQL), Second Generation RDF Query Language (SeRQL), and SPARQL which is a recursive acronym for SPARQL Protocol and RDF Query Language (Haase et al. (2004)). SPARQL is the most widely used query language due to its recommendation by W3C since 2008. Some of the OWL DL-based QLs are the Graphical query Language for OWL Ontologies (GLOO) (Fadhil and Haarslev (2006)) and its extension the Visual Query Language for OWL Ontologies (OntoVQL) (Fadhil and Haarslev (2007)), the Schema And Instance Query Language (SAIQL) (Kubias et al. (2007)), and SPARQL-DL (Sirin and Parsia (2007)), which is an extension to the RDF-based SPARQL. SPARQL-DL is aligned to SPARQL and hence provides interoperability of applications on the SW.

2.3 INTEGRATION

The main goal of integration is to enrich the KB with the unique facts extracted from multiple sources by removing duplicates and identifying new information related to an entity. This is achieved by finding the similarity between the entities. However, it is crucial to identify the similarity based on the semantics rather than the syntax (Saruladha et al. (2010)).

Many of the string similarity measures exist like character n-gram similarity (Kondrak (2005)), Leven-shtein Distance (Miller et al. (2009)), Jaro-Winkler measure (Xiao et al. (2008)), Jaccard similarity (Chaudhuri et al. (2006)), TF-IDF based cosine similarity (Salton and Buckley (1988)) and Hidden Markov Model-based measure (Miller et al. (1999)). However, these metrics are limited to measuring the similarity between strings syntactically but not semantically like synonyms using external knowledge sources like Wikipedia (Qu et al. (2018)), which provides a better semantic correlation between the entities.

To improve the quality of syntactic similarity measures, many of the machine learning-based methods like (Tsuruoka et al. (2007)) have been proposed. Even though these methods try to capture the semantic similarities between strings, they are limited to some pre-defined domains and need a huge set of training data for the effective capturing of semantics. Corpus-based methods like word embeddings and Knowledge-based methods like Wu & Palmer are also introduced by many works to find semantic similarity (Elavarasi et al. (2014); Islam and Inkpen (2008); Mihalcea et al. (2006); Xie and Liu (2008)). Recently, the authors in (Zhu and Iglesias (2018)) have conducted experiments for checking the semantic similarity using the knowledge and corpus-based methods. Based on their empirical study, corpus-based methods like embeddings do not consider various meanings of words, and when words have some relations like synonyms, the learned word vectors are not as accurate as knowledge-based methods. Moreover, for corpus-based methods when the training corpora changes, the similarities between the words are different due to changes in word vectors. Whereas, in knowledge-based methods, similarity scores are different only when the corresponding similarity metric changes as ontologies are normally stable and fixed. However, the corpus-based methods are more suitable to find the semantic relatedness or context between the two phrases based on the word co-occurrences or surrounding words and their frequency of occurrences in two phrases (Zhu and Iglesias (2018)). Hence it is critical to explore both corpus-based as-well-as knowledge-based methods to find similarities between semantically related entities.

2.4 VALIDATION

Validation of facts which can be done prior to or post construction of knowledge base, is least addressed in the existing woks. Figure 2.3 shows the taxonomy of works for checking the validity of facts. The existing works perform the post validation of facts either manually or automatically. Traditional methods like Never Ending Language Learning (NELL) (Carlson et al. (2010)), DBpedia (Lehmann et al. (2015)), YAGO2 (Hoffart et al. (2013)) and (Heindorf et al. (2016)) still follow manual means of checking the validity of facts whose cost is considerable. In recent years, the works like (Jia et al. (2019)), Knowledge vault (Dong et al. (2014)), (Liang et al. (2017); Nickel et al. (2011); Shi and Weninger (2016)) have attempted to automatically validate the knowledge base facts using embedding techniques. However, these models quantify the semantic correctness of facts from the knowledge graph. For instance, these models determine "whether the relationship r occurs between the entity pair $< h, t >$" and such

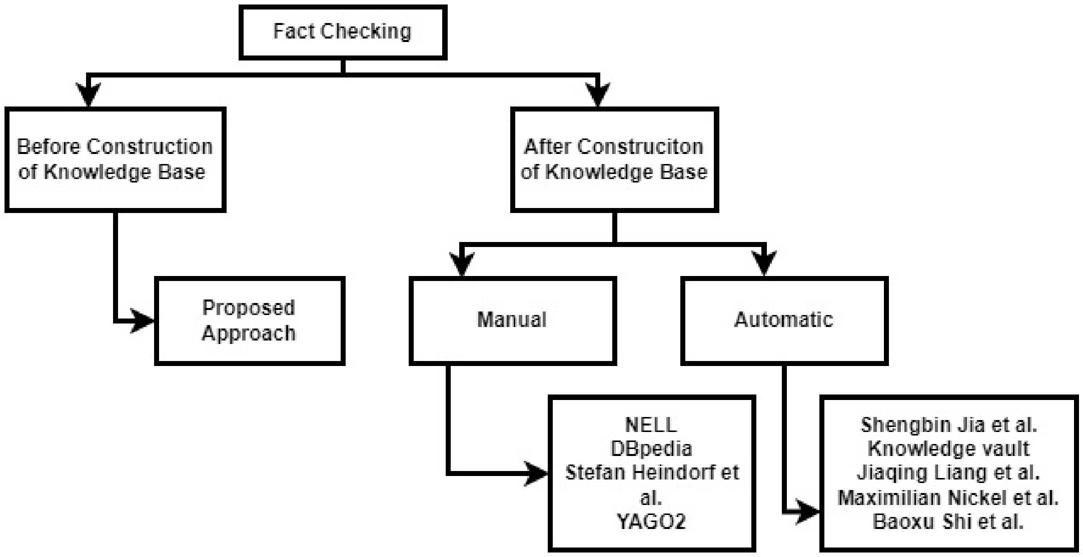


Figure 2.3: Taxonomy of methods for knowledge base facts validation

facts are inferred from the existing facts in the knowledge graph. However, any fake facts added into the knowledge base while extracting the information is not detected by these models. This is to be achieved before the construction of the knowledge base so that the knowledge base is enriched with validated facts. In this view, the proposed work for validating the knowledge base facts is the first attempt to check the credibility of facts before the construction of the knowledge base.

For the knowledge bases enriched with the facts extracted from news articles, checking for the credibility of facts can be considered as a problem of fake news detection. In recent years, researchers in the area of NLP and machine learning are attempting to develop models to alleviate the problem of fake news which is spread across the web by various users. Extant works to detect fake news are based on developing some powerful machine learning and deep learning models using various news datasets. For instance, in (Agarwal et al. (2020)) authors proposed a model using neural networks for fake news detection. Similarly, recent works like (Ghosh and Shah (2019); Gupta and Meel (2021); Kaliyar (2018); Reddy et al. (2020); Shu et al. (2019); Thota et al. (2018)) proposed their own model for detecting the fake news using different variants of machine and deep learning. However, these models are not targeted to train the knowledge base facts and hence not suitable to classify a given triple as fake or true from a knowledge base construction perspective.

2.5 RESEARCH CHALLENGES

1. **Impact of IE over the Integration:** Existing works considered IE and Integration as two independent problems. Hence, un-tagged and erroneously tagged entities generated while extracting the information and their impact like loss and redundancy of information over the final KB is ignored. For an instance consider the following two sentences:

   Sentence-1: Gurgaon Police on Wednesday night arrested Sampat Nehra, a member of the Lawrence Bishnoi gang, from Hyderabad.

   Sentence-2: Lawrence Bishnoi gang member arrested from Hyderabad.

   Named entity tags assigned for the above two sentences using NLTK, a well-known tool for NLP are as follows:

   NE tags for Sentence-1: PERSON:Gurgaon Police; PERSON:Sampat Nehra; ORGANIZA-TION:Lawre- nce Bishnoi; GPE:Hyderabad.

   NE tags for Sentence-2: GPE:Lawrence; PERSON:Bishnoi; GPE:Hyderabad.

   From the above tags, it can be observed that the entity “Lawrance Bishnoi” is labeled with different tags. Such erroneously tagged entities will be added twice into the knowledge base while merging entities from multiple sources and hence leads to redundancy. Similarly, the entity which indicates the time when the event happened i.e. Wednesday night is left un-tagged in sentence-1. Such un-tagged entities will not be added to the knowledge base and hence leads to loss of information.

2. **Explore the use of both corpus-based and knowledge based semantic similarity measures:** Based on the experimentation conducted by our work over several set of sentences, it is found that, the use of both similarity measures are necessary to identify the similarity of facts. To ensure the need for both the similarity measures, we experimented on around 300 sentences.

   Empirically it is found that the contextual meaning of the entities can be better

Table 2.2: Events and their description

Event-id	Event Description
E_1	Gurgaon police arrests key Lawrence Bishnoi gang member from Hyderabad
E_2	death at the bank of a river
E_3	ATM fraud at canara bank
E_4	10 year jail for raping minor in Mumbai
E_5	Woman drug officer shot in Kharar office; assailant dies of own bullet
E_6	Haryana Police Arrests Key Lawrence Bishnoi Gang Member From Hyderabad
E_7	fraud at canara bank
E_8	robbery at SBI bank
E_9	Woman officer shot dead at office in Punjab's Kharar
E_{10}	50-Year-Old Telangana Man Shot Dead In Florida Departmental Store

Table 2.3: Similarity scores between the events

Events	Contextual Similarity Score					Semantic Similarity Score				
	E_6	E_7	E_8	E_9	E_{10}	E_6	E_7	E_8	E_9	E_{10}
E_1	0.97	0.23	0.22	0.20	0.18	0.99	0.12	0.10	0.09	0.00
E_2	0.12	0.17	0.16	0.25	0.26	0.07	0.20	0.22	0.15	0.14
E_3	0.11	0.98	0.68	0.10	0.09	0.07	1.00	0.52	0.11	0.10
E_4	0.13	0.12	0.10	0.11	0.11	0.25	0.22	0.19	0.00	0.02
E_5	0.13	0.11	0.12	0.78	0.62	0.00	0.00	0.00	0.82	0.15

captured using corpus-based contextual similarity models like word embedding models due to vectors created by these models based only on the co-occurrences of entities in the events. However, these models do not provide a higher similarity score for semantically similar events due to the failure of measuring the synonymic relations. In contrast, the knowledge base algorithms provide a higher similarly score for semantically similar events provided they are contextually similar. Hence, it is necessary to consider the contextual meaning of the entities along with their synonymic relations to capture more semantics about the entities. And also, finding the contextual similarity before semantic similarity restricts the identification of similarity between only semantically related events and hence avoids the semantic similarity check between two semantically non-related events. In general, events that are not semantically related can not be semantically similar. To illustrate, tables 2.2 and 2.3 show the description of ten events and the similarity score between them using contextual and semantic similarity measures respectively. From the table 2.3 it can be observed that, for most similar events like E_1-E_6 and E_3-E_7, the contextual and semantic similarity scores are high and almost equal. Hence, it is feasible to find the contextual similarity before the semantic similarity. Another problem with using semantic similarity alone is in fixing the threshold value. This is overcome by fixing the threshold to the contextual similarity value. For example- semantically non-similar events like E_5-E_{10} and E_3-E_8 it can be observed that semantic similarity score is less than the contextual similarity score. Therefore, using either of the similarity measures alone does not capture complete semantics about the events. Consequently, both the similarity measures are adapted and empirically fixed a threshold of 0.5 for finding the contextual similarity.

3. **Utilization of images:** In an attempt to create domain-specific or independent KBs, a lot of works have explored the text data in extracting the facts from different sources. However, utilizing other modality data like images provide complete information about an entity in a single place. Using captions associated with the images along with high-level features for finding the semantic similarity of images is unexplored.

4. **Multi-Lingual IE**: The availability of news articles in various languages other than English, requires methods to extract the new facts from multi-lingual texts without any redundancies. Even though the problem of knowledge base construction is studied for decades, the knowledge base completion is studied only as a knowledge base refinement problem (Malaviya et al. (2020); Paulheim (2017); Pezeshkpour et al. (2020)). The knowledge base refinement methods try to complete the knowledge base by considering the facts internal to the knowledge base by inferring the new facts hidden inside the given knowledge base. However, these methods do not cover the external facts i.e. facts extracted from multiple sources while constructing the knowledge base. In this perspective, the knowl-

edge base completion can be treated as a problem of information extraction and integration, where the final knowledge base must be enriched with all the new facts extracted from multiple sources. In addition to the efforts to generate the knowledge base, several studies attempted to develop knowledge base completion models using cross-lingual projection of knowledge. However, these models require the presence of a knowledge base for both source and target language. Using the knowledge bases for both languages, the facts from the source language are projected with the target language for knowledge base completion. For instance, (Chen et al. (2017)) and (Klein et al. (2017)) developed a knowledge base completion model based on vector representation by representing the concepts in multiple languages in a unified vector space. But these models are not applicable in the absence of a knowledge base for a target language.

5. **Fact validation**: Existing machine learning models for fake news detection are not trained using triples. From the KBC perspective, there is a need for a machine learning model trained with enough triples to classify a given fact as either true or fake.

2.6 SUMMARY OF KNOWLEDGE BASE CONSTRUCTION SYSTEMS

The consolidated view of existing works divided based on their language support is shown in tables 2.4 and 2.5. The tables show the IE methods used by each work along with Domain, Source of Extraction, Language, and also other KB requirements like Modality of data extracted, support for Integration and Validation, and Knowledge Representation Format used.

Systems that extract the knowledge from sources in a single language are considered as Mono-Lingual systems and most of the systems extract the knowledge only from sources in the English language (Gerber et al. (2013)). In (Stern and Sagot (2012)) authors extract the facts from French-language news articles. Systems that extract the knowledge from sources of different languages are considered as Multi-Lingual systems. Most of the Multi-Lingual systems consider the sources in foreign languages (Rospocher et al. (2016)) and only (Akbik et al. (2016)) extracts the information from

Indian language Hindi along with foreign languages. However, these systems are based on either using language-specific processing tools or translating the entire documents into English.

The majority of the existing works extract and populate their KBs only with text modality data (Dasgupta et al. (2017)). Very few works like (Buitelaar et al. (2008)) used the image modality data along with texts for populating the KB. Accordingly, the KBs generated can be single modal text-based or multi-modal text and image-based. Storing images explicitly in the KB and integrating them with existing text entities is necessary for crime-related KBs because images act as additional evidence for an entity or event.

In contrast to the crime domain, a system called SOBA (Buitelaar et al. (2008)) generates the knowledge base for the sports domain by extracting and integrating the sports-related text and image entities from heterogeneous sources. Even though the system addressed the problem of removing the duplicates from the knowledge base, the loss of information while integrating is ignored. A system for an open intelligence platform like CAPER (Aliprandi et al. (2014)), considers multi-lingual texts belonging to 13 foreign languages along with English and also other modality data like audios and videos along with texts and images. However, the system ignored the effect of both duplicity and loss of information while integrating the data from multiple sources. Furthermore, none of the existing works validate the facts before they are incorporated into the knowledge base.

From the above-related works to the best of our knowledge, the system is unique to the domain of crime information extraction, integration, and knowledge base generation from multiple online newspapers without loss and duplicity.

Table 2.4: Features supported by mono-lingual systems

					Knowledge Base Requirements			
System	**Method of IE**	**Domain**	**Source of Extraction**	**Language**	**Other Modality data**	**Integration**	**Validation**	**Knowledge Representation Format**
CrimeProfiler(Dasgupta et al. (2017))	Stanford NER,Semi-supervised	Crime	News articles	English				Triples
SOBA(Buitelaar et al. (2008))	SProUT,Rule based	Sports	FIFA website	English	✓	✓		RDFS and FLOGIC
(Oramas et al. (2016))	Rule based,Un-supervised	Music	songfacts.com	English		✓		Triples
Artequakt(Alani et al. (2003))	Wornet+GATE+Ontology	Artists	Web	English		✓		Triples
PRISMATIC(Fan et al. (2012))	Frame+Un-supervised	Open	Web	English				Triples
NELL(Carlson et al. (2010))	Semi-supervised	Open	Web	English		✓		Triples
RdfLiveNews(Gerber et al. (2013))	Un-supervised,Supervised	Open	Web	English		✓		OWL
LODifier(Augenstein et al. (2012))	Wikifier,C&C parser,Boxer system	Open	Web	English				OWL
Storybase(Wu et al. (2015))	Un-supervised	Open News	Web	English				Triple
(Stern and Sagot (2012))	Entity based	Open News	Web	French				OWL
Knowledge Vault(Dong et al. (2014))	Distant Supervision	Open	Web	English		✓		Triples

Table 2.5: Features supported by multi-lingual Systems

					Knowledge Base Requirements			
System	**Method of IE**	**Domain**	**Source of Extraction**	**Language**	**Other Modality data**	**Integration**	**Validation**	**Knowledge Representation Format**
(Rospocher et al. (2016))	Semantic Role Labelling	Open News	Wikinews,FIFA world cup,Global Automotive Industry,Airbus A380 airplanes.	English,Spanish, Italian and Dutch		✓		Event Centric KGs
BOA(Gerber and Ngomo (2012))	Bootstrapping	Open	Web	English,German				Triples
New/s/leak(Wiedemann et al. (2018))	Using Polyglot tool	Open News	Web	40				Entity and Keyword graph
ZENON(Hecking and Schwerdt (2008))	GATE,Rule Based	Crime	Intelligence Reports	English,Deri		✓		Entity-Action-Network
POLYGLOTIE(Akbik et al. (2016))	Semantic Role Labelling	Open	Web	English, German, French, Spanish, Japanese, Chinese, Arabic, Russian and Hindi		✓		Triples
(Gamallo and Garcia (2015))	Rule based	Open	Web	English,Portuguese and Spanish				Triples

CHAPTER 3

PROBLEM DESCRIPTION

3.1 PROBLEM STATEMENT

The primary goal of this research is to create a bootstrap-based model for developing Crime Base, a knowledge base of crime entities and their relationships which is extracted and integrated from the text and image content of online news articles in English and Hindi language:

- without duplicity and loss of information.
- without/limited use of respective language processing tools and with minimum language translation efforts for articles other than the English language.
- with a validated set of facts.

3.2 OBJECTIVES

The primary goal is subdivided into the following objectives:

1. *Construction of a bootstrapping knowledge base:* This objective aims to construct a KB of bootstrapping triples by extracting crime-related text and image entities and their relations from English language news articles. To begin, a domain-aware crawler is required to build a corpus of crime-related English and Hindi news articles. Although this is possible with a dictionary, the investigation of articles is limited to the crime-related terms in the dictionary. As a result, it is

critical to automate the crawling process with a topic modeling technique such as Latent Dirichlet Allocation (LDA). However, to avoid false positives, it is also advised to use an external KB such as DBpedia. The crawler created in this manner should update the dictionary with more useful and unexpected keywords found in the articles. Secondly, propose a method for extracting entities and their relationships. Existing tools and techniques for the extraction of named entities have not thrown light on the effect of un-tagged and incorrectly tagged entities over the knowledge base. It is critical from the standpoint of knowledge base construction to minimize un-tagged and incorrectly tagged entities to reduce redundancy and information loss in the final knowledge base. Even though existing Open Relation Extraction (ORE) tools extract triples with high accuracy, they do not generate named entity tags. As a result, an algorithm for joint modeling of Named Entity Recognition and Relation Extraction must be Designed. Finally, the goal of the first objective is to use contextual as-well-as semantic similarity methods to perform semantic merging of extracted facts. Image captions are to be used to extract image entities from images, and both image captions and low-level image features are to be considered when integrating images from multiple news articles. The methods should concentrate on extracting and integrating entities and their relationships to maintain the knowledge base without duplication and loss of information.

2. *Development of a knowledge base completion framework:* Extracting facts, such as entities and relations, from unstructured sources is a critical step in any knowledge base construction process. Simultaneously, ensuring the completeness of the knowledge base by incrementally extracting new facts from various sources is required. The second goal is to create a framework for completing the knowledge base. Given a set of bi-lingual news articles from resource-rich source language like English and resource deficit target language like Hindi respectively. Knowledge base completion aims to extract the facts from target language news articles so that the knowledge base created using source language news articles also known as bootstrapping knowledge base is enriched with new facts available in

target language news articles. This is to be achieved by exploiting the redundancies available from source and target language news articles and without using the language-specific tools for target language news articles. To accomplish this, a clustering algorithm is vital, which explores the redundancy among the bi-lingual collection of news articles by representing the clusters with knowledge base facts unlike the Bag of Words representation used by the existing works. Within each cluster, the facts extracted from English language articles are to be bootstrapped to extract the facts from comparable Hindi language articles.

3. *Triple based fake news classification:* From a knowledge base development perspective, the third objective is to create a deep learning-based classifier for fake news detection. For any given triple extracted from a news article, the model should classify the triple as true or fake. This is achieved in three phases namely, data modeling, Investigation of a baseline model, and Implementation of the proposed Multi-layer Perceptron model. Data modeling involves the generation of numerical representations for each triple in the corpus using word embedding models. Word embedding can be either frequency-based or prediction-based. To capture word and document level features, and to reduce the vector dimensions, both frequency-based and prediction-based word embedding techniques are to be used. Frequency-based word embedding is to be achieved using TF-IDF and count vectorizer. Whereas, prediction-based word embedding is to be performed using GloVe and Word2Vec. Since it is a first attempt to train a deep learning model using triples for fake news classification, it is most essential to identify a baseline model for comparing the performance of the proposed model. This is to be achieved by training six machine learning classifiers namely Multinomial Naive Base (MNB), Passive-Aggressive (PA), Support Vector Machine(SVM), K-Nearest Neighbors(KNN), Logistic Regression(LR), and Random Forest(RF) using triples. The classifier that produces the high accuracy is to be used as the baseline for comparing the results obtained using the proposed deep learning-based Multi-layer Perceptron.

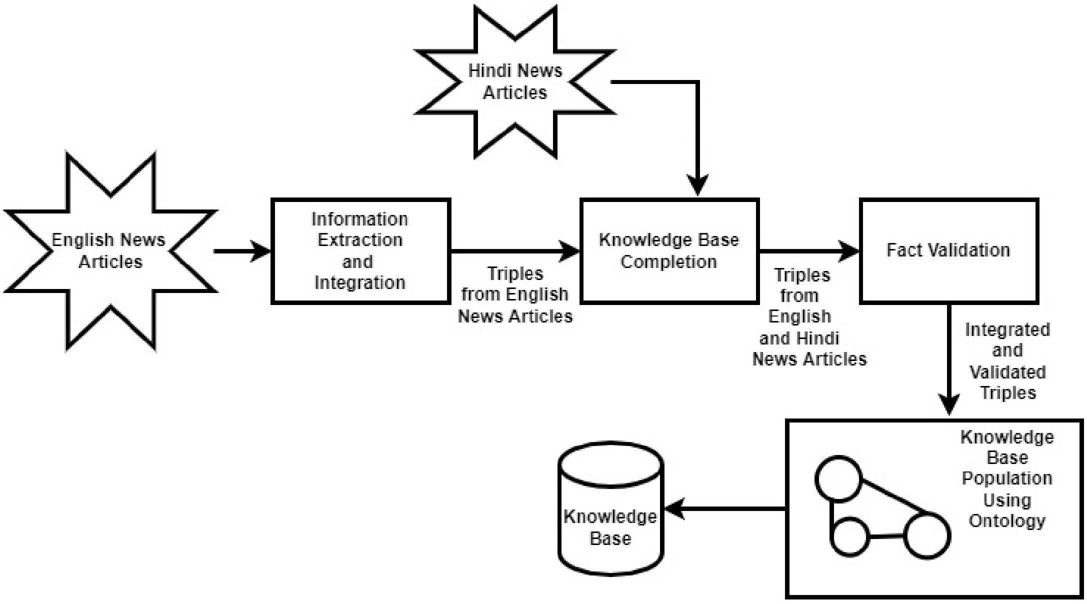

Hindi News Articles
English News Articles
Information Extraction and Integration
Triples from English News Articles
Knowledge Base Completion
Triples from English and Hindi News Articles
Fact Validation
Integrated and Validated Triples
Knowledge Base Population Using Ontology
Knowledge Base

Figure 3.1: Overview of the proposed model to create a crime knowledge base

3.3 SYSTEM ARCHITECTURE

The overall architecture of the proposed bootstrapping-based model for creating a crime knowledge base from online news articles is shown in the figure 3.1. Initially, the bootstrapping triples are extracted from English news using the proposed information extraction and integration techniques. Then the proposed knowledge base completion framework extracts the new triples from Hindi news articles by bootstrapping the triples extracted from English news articles. Thirdly, the triples extracted from English and Hindi news articles are checked for their credibility. Finally, the integrated and validated triples are populated to a knowledge base using an ontology, a knowledge representation model. Each of the stages is explained in detail in the following chapters.

CHAPTER 4

CONSTRUCTION OF A BOOTSTRAPPING KNOWLEDGE BASE

4.1 INTRODUCTION

The major challenge in developing a news KB from multi-lingual news articles is to extract the information from low-resourced language news articles. The existing works to achieve this, either using the language-specific tools or translating the low-resource language articles in their entirety to English. This is not feasible for the Indian languages, where the language-specific tools are neither available nor accurate. As a result, this is achieved in this work with the help of a Bootstrapping Knowledge Base (BKB) which is explained in chapter 5. This chapter deals with the construction of BKB.

A BKB is a knowledge base enriched with the facts extracted from high-resource language news articles and will be used to extract facts from low-resource language news articles without using language-specific tools. In this work, the bootstrapping KB is constructed by extracting the facts from English-language news articles. A primary challenge in any knowledge base generation system is in extracting and integrating the information coming from various sources. Extracting the named entities and their relations is a key requirement in extracting the information. Even though a lot of work has been done in extracting the information from English news articles, the problem of un-tagged and incorrectly tagged entities is not identified by the extant works. This leads to loss of information or redundancy in the final KB. For instance, when integrating information from multiple newspapers, un-tagged entities will not be added to the

KB, resulting in information loss. Similarly, erroneously tagged entities, even if they are semantically similar, will be added to the knowledge base multiple times, resulting in redundancy. Furthermore, relations of high quality can be extracted using open relation extraction techniques. However, these techniques do not assign any named entity tags to the entities. As a result, an entity and relation extraction mechanism that jointly extracts the entities and their relations by minimizing un-tagged and incorrectly tagged entities is important for a complete and non-redundant KB. It is also necessary to extract images to get the complete information about any entity, unlike the existing works that extract only crime-related text information from online news articles (Dasgupta et al. (2017)). Due to the unstructured nature of natural language texts, while integrating it is most important to check for the semantic similarity between entities and their relations rather than just checking for string-based similarity.

This chapter is aimed at developing a BKB of text and image entities and their relations extracted from online newspapers in the English language. The KB so developed is bootstrapped to extract the new facts from Hindi news articles. The proposed work focused on extracting and integrating entities and their relations with the desire to upkeep the knowledge base without duplicity and loss of information. Image captions are used to extract image entities and both image captions, as well as low-level image features, are considered when integrating images from multiple news articles.

The primary contributions of this chapter include:

- Proposed an algorithm for extraction of entities and their relations that minimizes loss and redundancy in the KB by reducing the un-tagged and incorrectly tagged entities using rule-based and DBpedia-based approaches.

- Explored the use of both contextual and semantic similarity measures for semantic integration of crime-related text entities, and image entities using high level and low-level features.

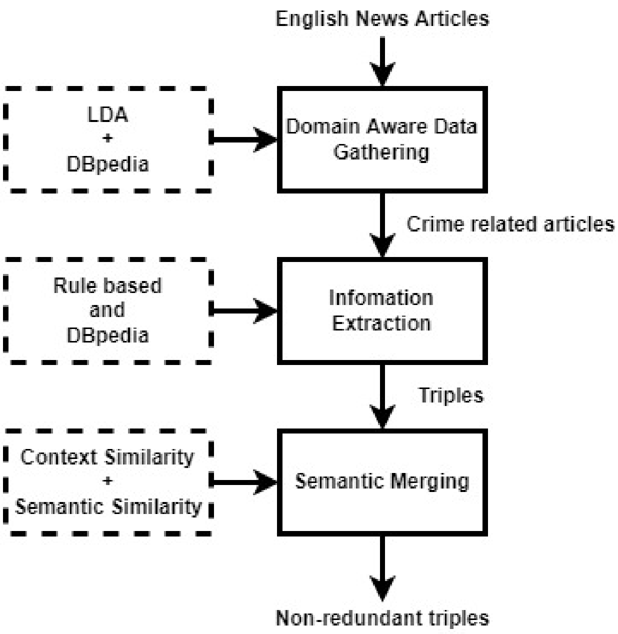


Figure 4.1: Overall Architecture

4.2 SYSTEM ARCHITECTURE

The overall architecture of the proposed solution to create a BKB is shown in figure 4.1. The proposed solution consists of three main stages: Data Gathering, Extraction of entities, and Semantic merging. The objective of the first stage is to develop a domain-aware crawler to build a corpus of crime-related news articles. The primary goal of the second stage, IE, is to assign named entities such as PERSON, ORGANIZATION, LOCATION, and so on to different segments of the text and extract their relationships by minimizing the un-tagged and incorrectly tagged entities. The purpose of semantic merging is to integrate the entities extracted from multiple sources to avoid duplication and to enrich the KB with new information about an entity. Each of the stages is discussed in detail in the following sections.

4.3 DATA ACQUISITION

In this work the topic modeling and knowledge base aided domain-aware crawler is developed which gathers and stores crime-related news articles from the online newspapers. The detailed process involved in gathering the crime-related texts and images via crawling of URLs specified by the user is shown in figure 4.2 and is explained in the following steps:

- The main-URL from which articles are to be scrapped is specified. For example, [1] is the main-URL to scrap the articles from Indian Express newspaper.
- For each of the articles present in the main-URL, the domain-aware articles are selected using article's headline, obtained from the sub-URLs. A sample of a sub-URL related to an article from Indian express newspaper is shown in [2].
- Selection of articles are initially bootstrapped by a bag of keywords, a corpus of 100 crime-related terms constructed by collecting the terms manually from the dictionary.
- For each news article, the terms present in the article's headline are compared with the keywords in the corpus. If the headline includes any of the keywords, the article is categorized as crime-related. Otherwise, Latent Dirichlet Allocation (LDA) (Blei et al. (2003)), a well-known topic modeling technique is applied to identify true positive or negative crime article based on the probability distribution score of the headline over a crime related topic. For the headline with a probability score less than the threshold, the article will be classified as true negative. Sometimes LDA will generate false positives due to improper distribution of the headline over a crime topic. This is overcome by extracting the keywords from the article and finding their thematic relatedness towards the crime domain using an external knowledge base, DBpedia. An article with a keyword belongs to DBpedia: crime category will be classified as true positive and corpus will be updated with the respective keyword. The proposed data gathering guided by DBpedia along with LDA also updates the corpus with more useful and unexpected keywords present in the articles.
- For each selected sub-URL, web pages are scraped by selecting only text related to the main article and the top image of the article.

[1] https://indianexpress.com/

[2] http://indianexpress.com/article/india/gurgaon-police-arrests-key-lawrence-bishnoi-gang-member-sampat-nehra-underworld-gangster-5207714/

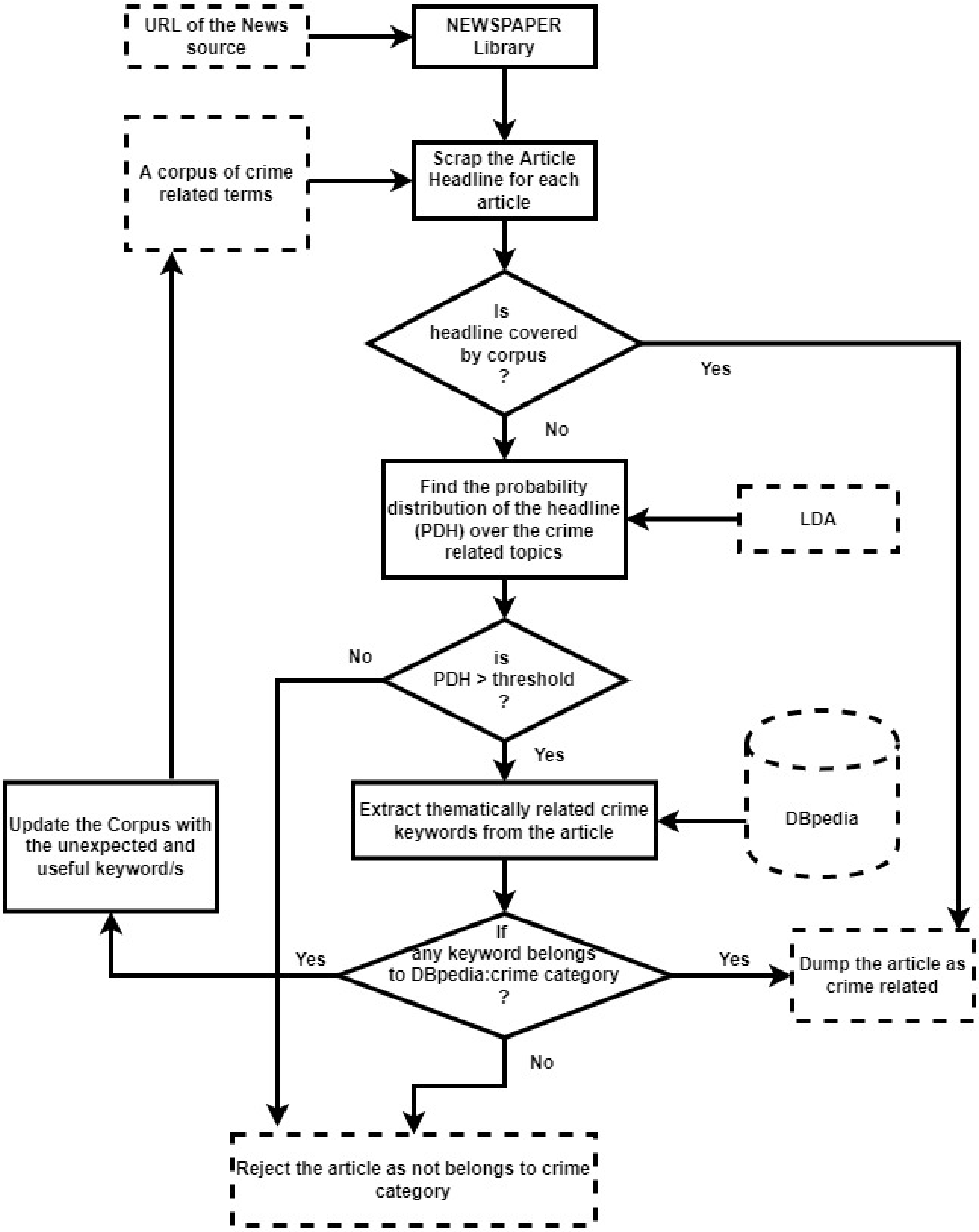

URL of the News source
NEWSPAPER Library
A corpus of crime related terms
Scrap the Article Headline for each article
Is headline covered by corpus ?
Yes
No
Find the probability distribution of the headline (PDH) over the crime related topics
LDA
No
is PDH > threshold ?
Yes
DBpedia
Update the Corpus with the unexpected and useful keyword/s
Extract thematically related crime keywords from the article
Yes
If any keyword belongs to DBpedia:crime category ?
Yes
Dump the article as crime related
No
Reject the article as not belongs to crime category

Figure 4.2: Topic Modeling and Knowledge base aided data acquisition

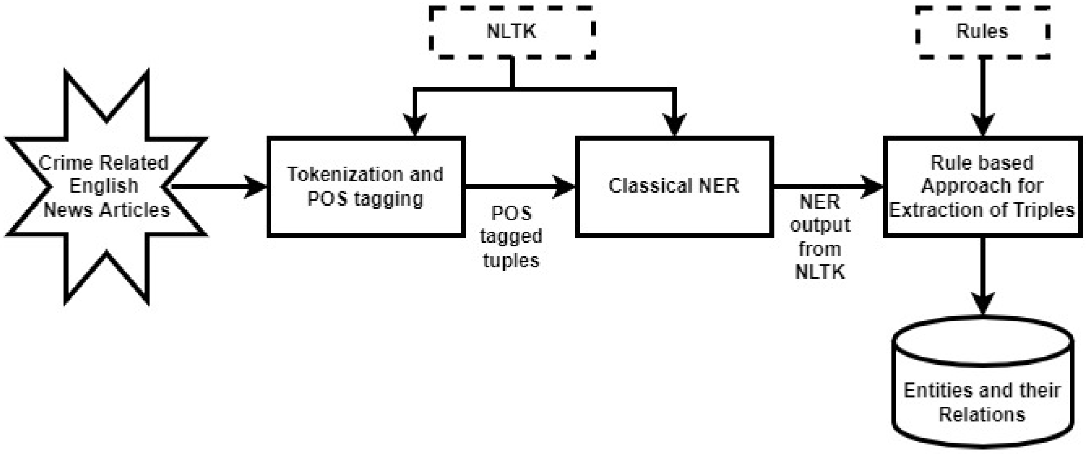

NLTK
Rules
Crime Related English News Articles
Tokenization and POS tagging
POS tagged tuples
Classical NER
NER output from NLTK
Rule based Approach for Extraction of Triples
Entities and their Relations

Figure 4.3: Rule based Generation of Entities and Relations

- Scraped text and image content is dumped to an output file for further processing.

4.4 INFORMATION EXTRACTION

The primary goal of any information extraction system is in assigning the named entities to different segments of the text and extracting their relations (Popov et al. (2003)). From the KBC perspective, it is essential to minimize the un-tagged and erroneously tagged entities generated while extracting the entities and their relations, to reduce redundancy and loss of information respectively. This is accomplished in this work through the use of rule-based and DBpedia-based approaches and is explained in detail in the sections that follow.

4.4.1 Rule based Approach

An algorithm is proposed to extract information from texts based on the combination of NLP and rule-based techniques. To extract entities and relations, some hand-crafted rules are applied over the Parts-of-Speech (POS) and NER tags generated by NLTK. This improves the performance of named entities obtained from NLTK in terms of the reduced number of erroneously tagged and untagged texts. The process involved in the generation of entities and their relations by the proposed approach is shown in figure 4.3 and is explained in the following steps.

Tokenization and POS Tagging The texts gathered for each article during data acquisition step are tokenized into sentences and words. Tokenized words of each sentence are tagged with POS tags using NLTK POS tagger. POS tags generated for two sample sentences crawled from Indian express newspaper [3] are shown below. Information about meaning for each tag can be obtained from (Loper and Bird (2002)).

Sentence-1: In a major breakthrough, Gurgaon Police on Wednesday night arrested Sampat Nehra, a member of the Lawrence Bishnoi gang, from Hyderabad.

POS tags: In/IN,a/DT,major/JJ,breakthrough/NN,,/,,Gurgaon/NNP,police/NN,on/IN, Wednesday/NNP, night/NN,arrested/VBD,Sampat/NNP,Nehra/NNP,,/,,a/DT,member/NN, of/IN,the/DT,Lawrence/NNP,Bishnoi/NNP,gang/NN,,/,,from/IN,Hyderabad/NNP,./.

Sentence-2: According to police, 28-year-old Nehra was the gang's sharp-shooter, and entered the underworld through the route of student politics.

POS tags: According/VBG,to/TO,police/NN,,/,,28-year-old/JJ,Nehra/NNP,was/VB D,the/DT,gang/NN,'/NNP,s/VBD,sharp-shooter/NN,,/,,and/CC,entered/VBD,the/DT,un derworld/NN,through/IN,the/DT,route/NN,of/IN,student/NN,politics/NNS,./.

Named Entity Recognition and Relationship Extraction Based on the POS tags obtained from the previous step, meaningful entities and their relations are identified with the help of $\langle subject - verb - object \rangle$ rule. The proposed entity and relation extraction using rule-based approach is shown in Algorithm 1.

Unlike many works that extract the relations after extracting the entities (Dasgupta et al. (2017)), the proposed work follows open relation extraction (Zouaq et al. (2017)) which first extracts the relations to identify the related entities properly and assign tags to un-tagged terms to avoid loss of information while integrating data from multiple sources. For example, the NER tags assigned by NLTK for sentence-2 is shown below:

According/VBG,to/TO,police/NN,,/,28-year-old/JJ,(PERSON Nehra/NNP),was/VB D,the/DT,gang/NN,'/NNP,s/VBD,sharp-shooter/NN ,/,and/CC,entered/VBD,the/DT,und erworld/NN,through/IN,the/DT,route/NN,of/IN,student/NN,politics/NNS.

[3] http://indianexpress.com/article/india/gurgaon-police-arrests-key-lawrence- bishnoi-gang-member-sampat-nehra-underworld-gangster-5207714/

Algorithm 1: Rule based Algorithm for Entity and Relation extraction

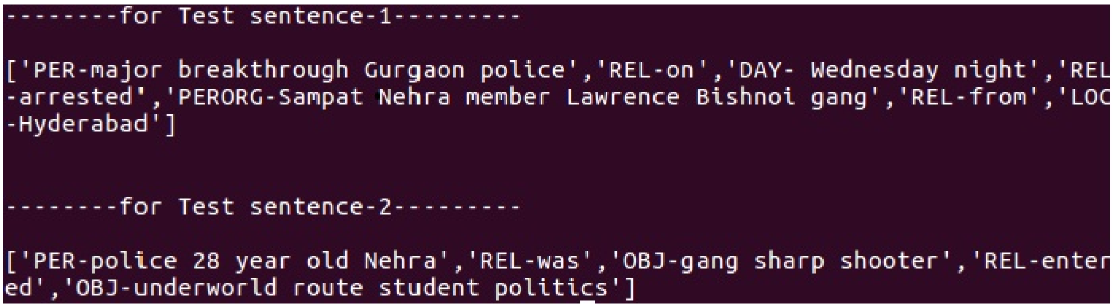
```
Input: $P$: Set of POS tagged tuples, $R$: Set of Rules
Output: $T$: Set of tuples that represent entities and their named entity tags
foreach tuple $t$ of $P$ do
    foreach Relation term $RE$ of $t$ do
        if $RE$ has non-empty left and right part then
            Divide the sentence at the $RE$
            $Subject \leftarrow Left(RE)$
            $Object \leftarrow Right(RE)$
    Apply the rules from $R$, assign named entities and generate the tuples in the
      form of ['key- value'] pair into $T$
```

Here, only "Nehra" is recognized as an entity and tagged with PERSON. Other attributes like the age of the person is not recognized and hence will not be added to the knowledge base if not considered for tagging.

To address the issue, initially, the relations are extracted by applying the following rules and are tagged with REL:

- Any term with POS tag like 'VB', 'VBD', 'VBN', 'VBG', 'VBZ' and 'VM' is considered as a verb and is a primary candidate for relation. Similarly, terms "on", "from" and "at" are also selected as relations.
- Any verb followed by a term with POS tag as 'IN' are considered together as a single relation. For example "of" in "accused of" is taken along with the verb "accused" to represent it as a relation.
- Two or more continuous verb terms are considered as a single relation.

Once the relations are extracted, terms from the English stop word list which are not identified as a relation and delimiters from the POS tagged list are removed before proceeding to the next step. Then each sentence is divided based on relationships as a pivot in such a way that it has non-empty left and right parts and is considered as 'subject' and 'object' respectively. The dividing process will continue until there is no relation found in each part. The rest of the work is in assigning the possible named entity tags to the words in the subject and object part of the sentences which is done by



```
--------for Test sentence-1---------

['PER-major breakthrough Gurgaon police','REL-on','DAY- Wednesday night','REL
-arrested','PERORG-Sampat Nehra member Lawrence Bishnoi gang','REL-from','LOC
-Hyderabad']


--------for Test sentence-2---------

['PER-police 28 year old Nehra','REL-was','OBJ-gang sharp shooter','REL-enter
ed','OBJ-underworld route student politics']
```

Figure 4.4: NER tagged tuples generated by the rule based approach

applying the hand-crafted rules listed in table 4.1.

NER tags assigned by the proposed method for the sample sentences is shown in figure 4.4. The output is generated in the form of tuples that include key-value pairs to represent named entity tag and the corresponding name of the entity. The proposed rule-based approach identifies seven types of named entities, including PERSON, ORGANIZATION, LOCATION, DAY, SUBJECT, OBJECT, and PERORG.

Even though the approach tags un-tagged entities to avoid information loss, tags such as SUBJECT, OBJECT, and PERORG are not meaningful and thus do not properly indicate the type of an entity. Furthermore, separate rules must be written to identify an additional number of named entities, which comes at a high cost. To address these issues, one more approach based on an external knowledge base is proposed and detailed in the following section.

4.4.2 DBpedia based Approach

In contrast to rule-based approach, an external KB such as DBpedia is regarded as a source of fact extraction in this approach. Common background knowledge for fact extraction helps to assign common tags to the entities independent of sentence structures. Even though there exist Open Relation Extraction (ORE) tools to extract the relations, we can't extract the relations related to the main event like $arrested\_on$, $arrested\_from$, and so on. Hence, this work proposed an algorithm for joint modeling of the NER and RE using DBpedia as common background knowledge and is shown in Algorithm 2.

The suggested algorithm accepts an English sentence as input. The sentences are

Table 4.1: Rules for NER tagging

Rules	Description
$(\text{“}from\text{”} \mid \text{“}at\text{”} \mid \text{“}in\text{”}\ (Noun)+)$ $\rightarrow LOCATION$	One or more Noun terms preceded by “from” or “at” or “in” clause is labelled as LOC(LOCATION)
$(\text{“}on\text{”}\ (Noun)+) \rightarrow DAY$	One or more Noun terms preceded by “on” clause is labelled as DAY
$(((Noun) * (Adjective) * (Noun)*) * Verb)$ $\rightarrow PERSON/ORGANIZATION/PERORG$	Any combination of Noun and adjective terms succeeded by verb are labeled with either (PER) PERSON or (ORG) ORGANIZATION based on the label provided by NLTK to any of the noun phrases. If both PERSON and ORGANIZATION terms are present, then the combination is labeled as PERORG. If none of the labels are assigned by NLTK, it is treated as a subject and is represented as SUB.
$(Verb((Noun) * (Adjective) * (Noun)*)*)$ $\rightarrow PERSON/ORGANIZATION/PERORG$	Similar to the previous pattern but the pattern is preceded by verb. Here, If none of the labels assigned by NLTK, it is treated as object and is represented as OBJ.

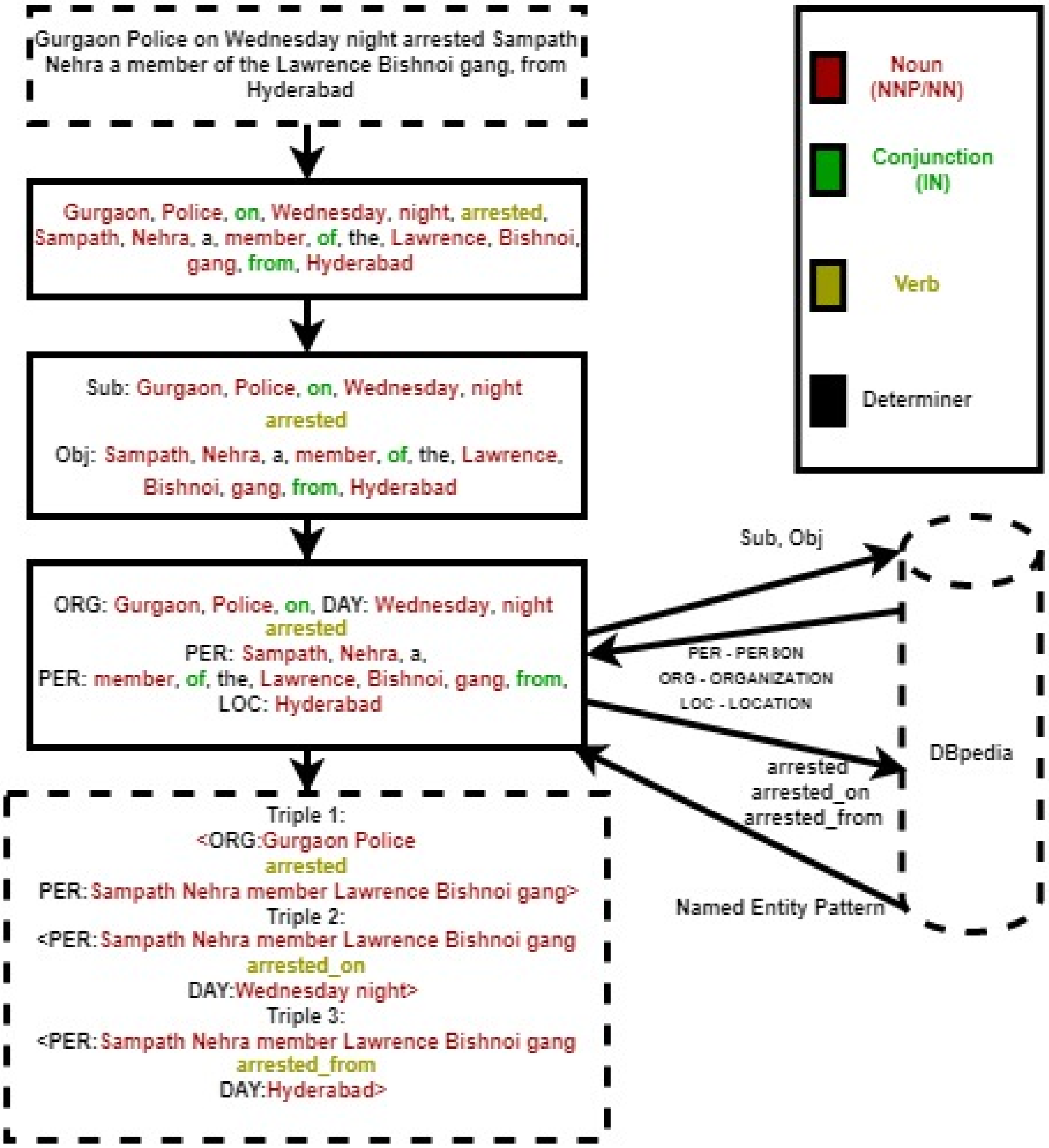

Gurgaon Police on Wednesday night arrested Sampath Nehra a member of the Lawrence Bishnoi gang, from Hyderabad
Gurgaon, Police, on, Wednesday, night, arrested, Sampath, Nehra, a, member, of, the, Lawrence, Bishnoi, gang, from, Hyderabad
Sub: Gurgaon, Police, on, Wednesday, night
arrested
Obj: Sampath, Nehra, a, member, of, the, Lawrence, Bishnoi, gang, from, Hyderabad
ORG: Gurgaon, Police, on, DAY: Wednesday, night
arrested
PER: Sampath, Nehra, a,
PER: member, of, the, Lawrence, Bishnoi, gang, from,
LOC: Hyderabad
Triple 1:
<ORG:Gurgaon Police
arrested
PER:Sampath Nehra member Lawrence Bishnoi gang>
Triple 2:
<PER:Sampath Nehra member Lawrence Bishnoi gang
arrested_on
DAY:Wednesday night>
Triple 3:
<PER:Sampath Nehra member Lawrence Bishnoi gang
arrested_from
DAY:Hyderabad>
Noun (NNP/NN)
Conjunction (IN)
Verb
Determiner
Sub, Obj
PER - PERSON
ORG - ORGANIZATION
LOC - LOCATION
arrested
arrested_on
arrested_from
Named Entity Pattern
DBpedia

Figure 4.5: Illustration of triple extraction with an example using DBpedia based Approach

Algorithm 2: DBpedia based Algorithm for Entity and Relation extraction

```
Input: $S$ (A Sentence in English)
Result: $T$ (A set triples along with named entity tags)
if $S$ is in passive voice then
    Convert $S$ to active voice
end
else
    Find the POS tags for $S$ using NLTK
    Identify the relational term $RE$ using the POS tag for verb
    $Subject \leftarrow Left(RE)$
    $Object \leftarrow Right(RE)$
    Query DBpedia for $n - gram$ words in $Subject$ and $Object$ where $n >= 1$
    Find the longest $n - length$ sequence for which we can find the DBpedia
      named entity tag
    Find all the $verb\ forms$ of $RE$ using $IN$ tagged terms
    while verb form do
        Query DBpedia using the $verb\ form$ for the named entity patterns
        Search for the pattern and extract the relations
    end
end
```

converted to active voice exclusively for clarity. To begin, NLTK is used to find POS tags for the sentence. The sentence's relational term, such as $arrested$, is then recognized using the POS tagger's VB tagged term. The $subject$ and $object$ of a sentence are the left and right phrases of the relational term. To extract named entity tags, DBpedia is queried using the SPARQL Protocol and RDF Query Language (SPARQL) for $n - gram$ terms in the subject and object. A named entity tag will be assigned to the longest $n-gram$ word sequence. The IN tagged phrases like on, for, $from$, and so on are used to identify all the verb forms of relational terms like $arrested\_on$. The named entity pattern for each of the verb forms is extracted by querying DBpedia once again. Finally, a triple is derived from the sentence based on the availability of the pattern. Using background knowledge for fact extraction in this way aids in the assignment of common tags to entities irrespective of sentence structures. Figure 4.5 illustrates the extraction of triples for an example sentence.

Similarly, the entities associated with the images are extracted by applying the proposed algorithm to the caption associated with them.

4.5 SEMANTIC MERGING

The goal of semantic merging is to integrate the events and entities extracted from multiple sources to avoid duplication and to enrich the KB with the new information about an entity or event (Dragos (2013)). This is achieved by checking correlation or similarity between the entities of two events obtained from different newspapers so that, the final knowledge base is enriched with a single instance of the correlated events and entities. Figure 4.6 shows the proposed method for semantically merging the two events using contextual as-well-as semantic similarity measures.

Due to the extraction of image data along with texts, the similarity between the entities of similar types is considered at two phases i.e. similarity between text and image entities respectively. The subsections that follow discuss the similarity measures used by each phase.

4.5.1 Similarity between text entities

In this work, the power of contextual as-well-as semantic similarity measures is explored to find the similarity between the entities of two events. The empirical study conducted to ensure the need for both the similarity measures is explained in section 2.5.

Contextual Similarity Measure: Here, Word2Vec, a predictive-based word embedding model is used to represent the events in a dense and low dimensional vector space due to its high performance in many applications (Zhu and Iglesias (2018)). The vector for each entity represents its description of how it occurs in context with other entities in an event. Hence, if two entities are used similarly in two different events, the chances of obtaining a similar vector representation for those entities are more. For example, if a relation “killed” is related by two-PERSON entities in an event, its probability of being related by the same or different person entities in other events is high. After mapping entities into the vector space, their similarity is computed using standard cosine similarity measure (Salton and Buckley (1988)). The contextual similarity is computed before semantic similarity. This restricts the identification of similarity between only semantically related events and hence avoids the semantic similarity check between two

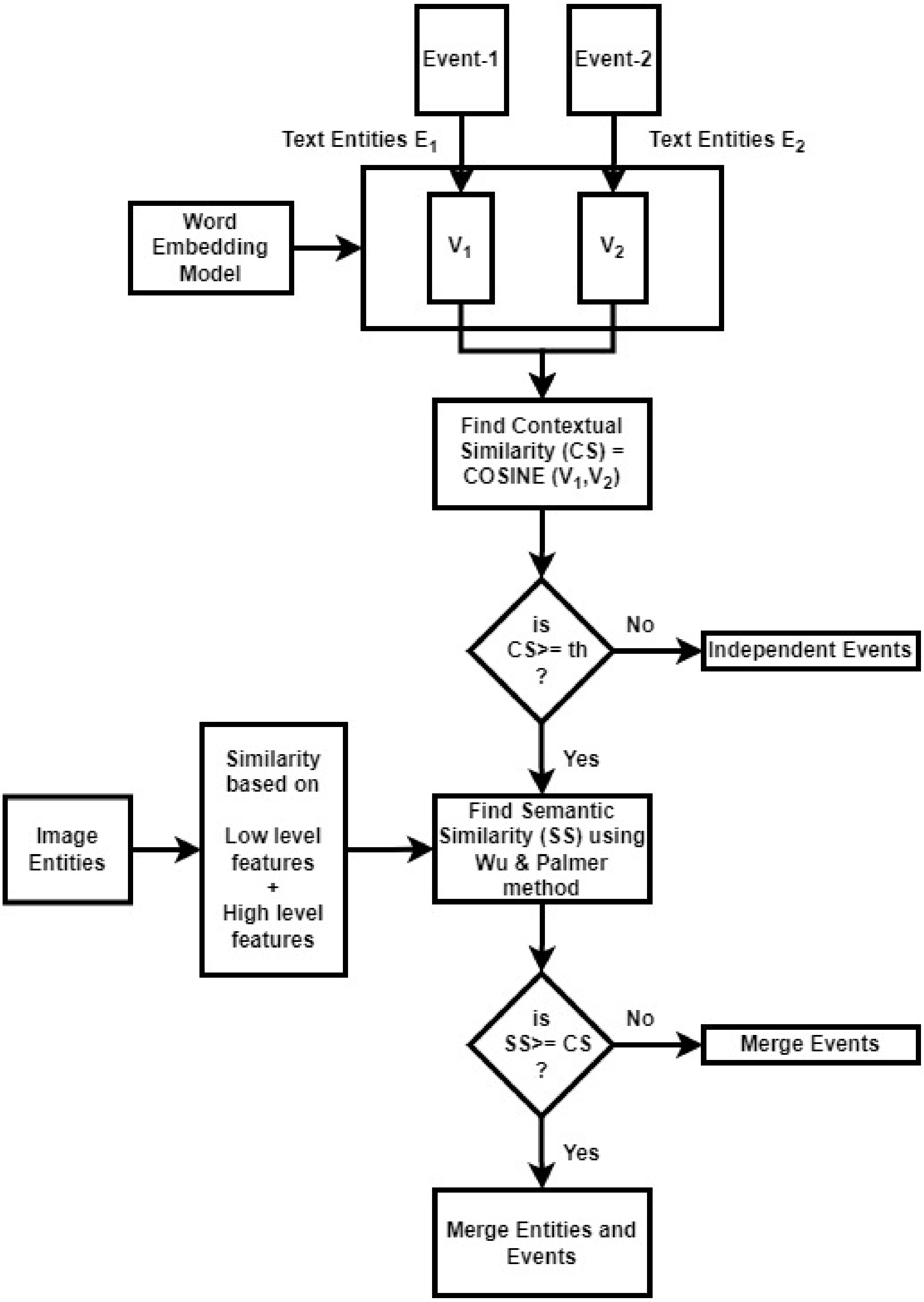

Event-1
Event-2
Text Entities E_1
Text Entities E_2
Word Embedding Model
V_1
V_2
Find Contextual Similarity (CS) = COSINE (V_1,V_2)
is CS>= th ?
No
Independent Events
Yes
Image Entities
Similarity based on Low level features + High level features
Find Semantic Similarity (SS) using Wu & Palmer method
is SS>= CS ?
No
Merge Events
Yes
Merge Entities and Events

Figure 4.6: *Semantic merging of events and entities*

semantically non-related events. Hence, it is feasible to find the contextual similarity before the semantic similarity. Events whose contextual similarity score is less than a threshold are treated as independent. Otherwise, they are examined for their semantic similarity by using the following method.

Semantic Similarity Measure: In general, events that are not contextually similar can not be semantically similar. Moreover, the difficulty in using semantic similarity alone is in fixing the appropriate threshold value to prove the semantic similarity of entities. This is overcome by setting the threshold to the value of the contextual similarity score. The capability of Word2Vec models lies in handling large corpus and trains the word vectors efficiently. However, due to the use of only the word sequences or co-occurrences of words for training the word vectors, the Word2Vec model fails to handle words with synonymous and hierarchical relations precisely (Zhu and Iglesias (2018)). As a result, Wu & Palmer method, a knowledge base algorithm to achieve semantic similarity is adapted by using the information available from a well-known ontological repository, WordNet. The semantic similarity measure computes the similarity based on the depth of the two synsets of the words in WordNet taxonomy along with the depth of the Least Common Subsumer (LCS) (Wan and Angryk (2007)).

Similarity between two Events The contextual or semantic similarity between two events say E_1 and E_2 is calculated by the weighted sum of the similarity between different entities from the two events as follows:

$$sim(E_1, E_2) = \frac{\sum_{i=1}^{n} w_i \, * \, sim(E_{1i} \, , \, E_{2i})}{min(|E_1|, |E_2|)} \tag{4.1}$$

where n is the number of entity types, w_i represents the weight of an i^{th} entity type and is empirically fixed based on the contribution of a named entity type in computing the similarity. E_{1i} and E_{2i} represents the i^{th} entity of the first and second events respectively.

Concerning contextual similarity, events whose contextual similarity score is less than a threshold are treated as independent. Otherwise, they are examined for their semantic similarity. Finally, two events are said to be semantically similar if the similarity score is greater than a threshold i.e. contextual similarity score and in such case, KB is enriched with a single instance of the correlated events. Otherwise, two events are decided to be only semantically related but not similar and hence are merged into a single event without merging their respective entities.

4.5.2 Similarity between Image entities

In this work, two image entities are compared for semantic similarity by matching low-level features as well as high-level captions. For the former case, the Structural Similarity Index(SSIM) method (Wang et al. (2004)) is used that perceive changes in the structural information of the image by comparing its local regions. Small sub-samples of the whole image of two images are compared and a similarity score is obtained by calculating the structural similarity index on various windows of the same sizes.

For finding the similarity of images using high-level captions, a vector-based method is proposed that interprets the images semantically rather than just interpreting images based on low-level features like color, shape, etc.

For the two images to be compared, a two dimensional vector V of captions and named entity chunks identified for the captions are formed. An entry $V[i, j]$ into the vector is equal to 1, if the i^{th} caption includes the j^{th} named entity, otherwise is equal to 0. Final similarity score based on the high level captions is calculated as:

$$caption\_score = \frac{\sum_{i \in caption} \sum_{j \in NamedEntity} V[i][j]}{Size(V)} \tag{4.2}$$

The final similarity score for two image entities i_1 and i_2 is the average of the similarity score obtained from low-level features and high-level captions and is calculated as:

$$image\_similarity(i_1, i_2) = \frac{SSIM(i_1, i_2) + caption\_score(i_1, i_2)}{2} \tag{4.3}$$

The extracted and integrated text and image entities from various news sources will be used to build a knowledge base known as knowledge base population using a knowledge representation model, as explained in section 4.6. Image entities are associated with each of the events via a relevance metric, which is also explained in section 4.6.

4.6 KNOWLEDGE BASE POPULATION

The goal of the knowledge Base population is to construct a knowledge base with the validated and non-redundant facts extracted and integrated from various crime-related online news articles. This is achieved by using a knowledge representation model (Trentelman (2009)) and adding the entities and their relations as an instance of the model. Ontology, one of many knowledge representation models such as distributed, symbolic, probabilistic, and rule-based, is now widely used in many areas such as artificial intelligence, biomedical informatics, semantic web, system engineering, forensic data analysis, and information architecture. Knowledge base population using ontology is performed by mapping the triples to equivalent components of ontology developed using Web Ontology Language (OWL). The procedure to generate the knowledge base using ontology is shown in Algorithm 3. The significance of the work is in representing image data along with text data in the knowledge base which provides complete details of an entity in a single place. The ontology structure used to populate the knowledge base is shown in the figure 4.7. Here the knowledge base is generated using Owlready which is a Python 3 library for working with OWL 2.0 ontologies [4]. It provides API calls to create ontologies, load existing ones as Python objects, modify them, and save them as OWL XML files. The triples generated for each sentence are treated as a tuple and are considered to include the entities into the knowledge base. Corresponding to each tuple, an event is created with an event id represented as $event\_article\ number\_sentence\ number$ to identify the events. Edges are added between the event and the entity objects. These edges are labeled according to the nature of the entity like location, organization, etc. Relation entities are also connected to other entities that participated in the relation. In the ontology, image entities are linked with each of the events if the images are associated with any of the respective entities belonging to the events and are labeled with the respective Relevance Measure (RM). RM represents the relevance of the image with an event and is calculated as shown in equation 4.4:

$$RM = \frac{(Number\ of\ entities\ of\ an\ Image\ matched\ with\ entities\ of\ an\ Event)}{(Total\ number\ of\ entities\ associate\ with\ the\ Image)} \tag{4.4}$$

To envisage the knowledge base in OWL format, a knowledge base is visualized

[4] https://pythonhosted.org/Owlready/

Algorithm 3: Algorithm for Knowledge Base Population

```
Input: $T$: List of NER tagged tuples
Output: Knowledge Base in OWL format
foreach Tuple $t$ in $T$ do
    $Event = NewEvent(RandomEventID)$
    $REL\_FOUND \leftarrow 0$
    $PREV\_ENTITY \leftarrow NULL$
    $count \leftarrow 0$
    foreach Elelment $e$ of a Triple in $t$ do
        if $e$ is Relation then
            Event.action.append(Element name)
            $REL\_FOUND \leftarrow 1$
            $REL\_item \leftarrow Element$
        if $e$ is Location then
            Event.location.append(Element name)
        Other entity types are also added like LOCATION except Image
        if $e$ is Image then
            foreach Triple in NER tagged tuple obtained from Image caption do
                foreach Element1 of the Triple do
                    if ((Element1 name) is part _of or equal _to (Any Element))
                      or vice versa then
                        $count++$
            if $count > 0$ then
                $RM = count/|item1|$
                Event.Image.append(Element1.RM)
```

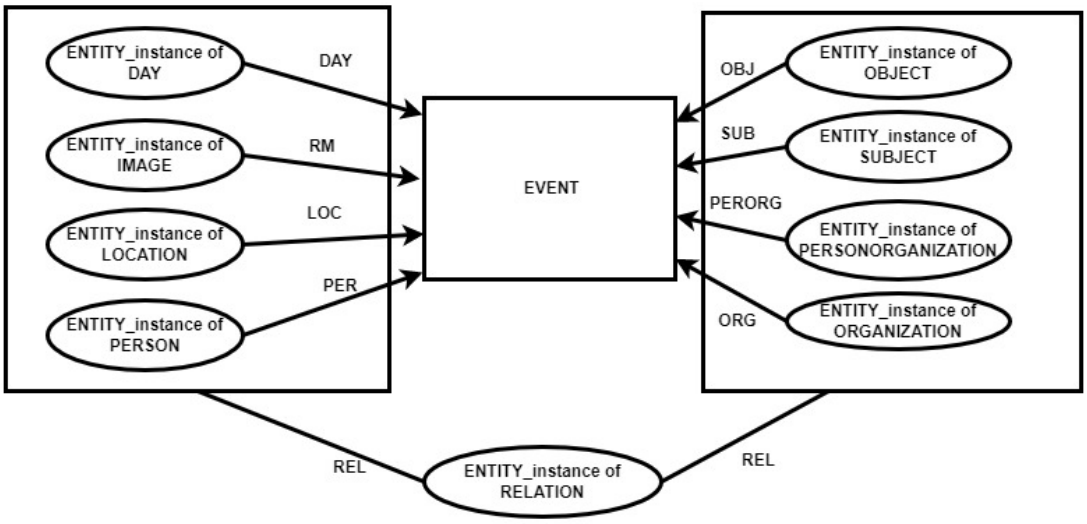

ENTITY_instance of DAY
ENTITY_instance of IMAGE
ENTITY_instance of LOCATION
ENTITY_instance of PERSON
DAY
RM
LOC
PER
EVENT
OBJ
SUB
PERORG
ORG
ENTITY_instance of OBJECT
ENTITY_instance of SUBJECT
ENTITY_instance of PERSONORGANIZATION
ENTITY_instance of ORGANIZATION
REL
ENTITY_instance of RELATION
REL

Figure 4.7: *Ontology structure*

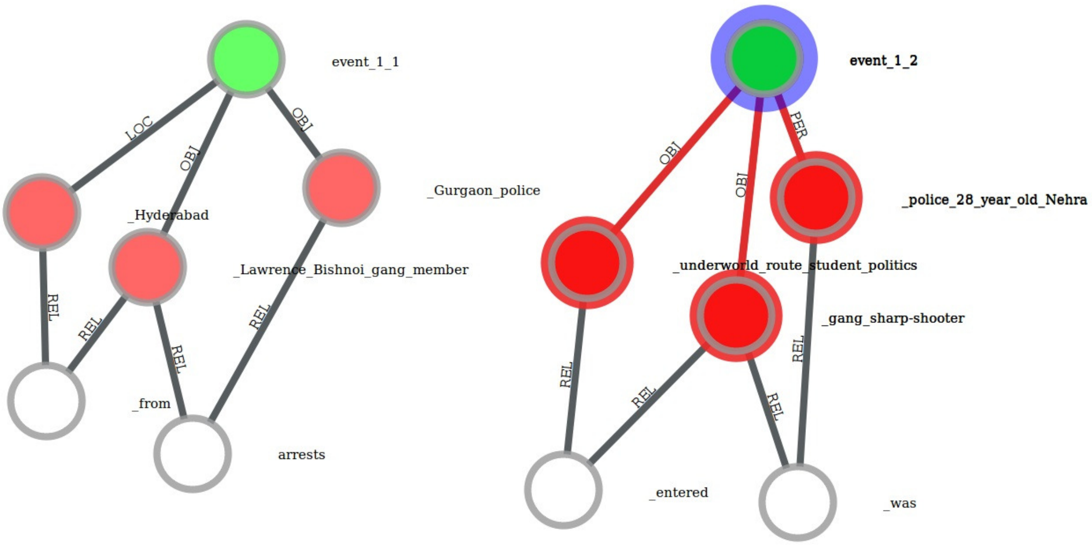


Figure 4.8: *Entity graph without image*

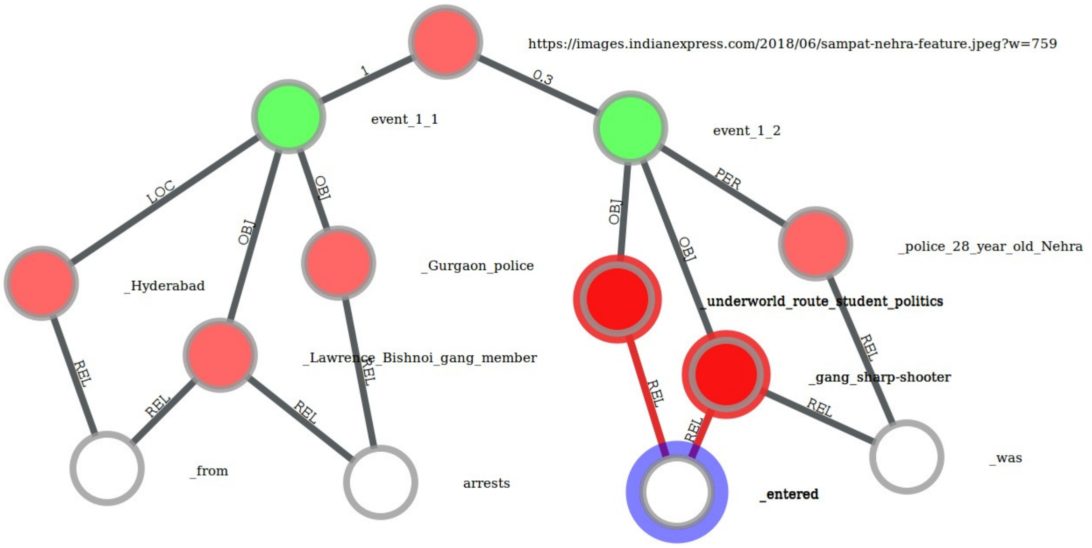

https://images.indianexpress.com/2018/06/sampat-nehra-feature.jpeg?w=759
1
0.3
event_1_1
event_1_2
LOC
OBJ
OBJ
PER
OBJ
OBJ
_Hyderabad
_Gurgaon_police
_police_28_year_old_Nehra
_underworld_route_student_politics
_Lawrence_Bishnoi_gang_member
_gang_sharp-shooter
REL
REL
REL
REL
REL
REL
REL
REL
_from
arrests
_entered
_was

Figure 4.9: *Entity graph with image*

as an interactive graph with the help of a graph tool [5] with all the entities displayed as nodes and relationships as edges between them. Knowledge graph so generated presents the information in a concise and interactive form. The user can zoom in/out and also drag the nodes to filter out only the entities he/she is interested in. Figures 4.8 and 4.9 illustrate the interactive graph generated for an OWL file related to two events. Figure 4.8 shows the graph for the events without an image entity being added to it. To illustrate the graph with an image entity, an image with the caption “Bishnoi key gang member Sampat Nehra arrested by Gurgaon police”, is considered. Figure 4.9 shows the updated graph with the image as an entity added to the graph in figure 4.8. Since the entities extracted from the caption of the image are matched with all the entities of $event\_1\_1$ and a single entity of $event\_1\_2$, the image is associated with $event\_1\_1$ and $event\_1\_2$ with RM 1 and RM 0.3 respectively. Here, Green, Red, and White nodes represent Event, Entity, and Relation respectively.

4.7 EXPERIMENTAL ANALYSIS

This work considers three prominent newspapers namely, *Indian Express, Times of India, and Deccan Chronical* which have articles available online to develop a corpus of English news articles. The corpus includes the data collected from Jan 2018 to

[5] https://graph-tool.skewed.de/

Jun 2018. The following sections describe the experimentation and evaluation results for data gathering, entity extraction, and semantic merging for bootstrapping knowledge base construction.

4.7.1 Evaluation of topic modeling and knowledge base aided data gathering

The proposed method for data gathering using topic modeling and DBpedia has achieved a significant improvement in identifying the true positive articles and also in filtering the true negative and false positive articles compared to the selection of articles based only on a corpus of crime-related terms. The experiment is conducted by training the LDA model using ABC News headlines dataset [6]. To illustrate the advantage of the proposed approach, a corpus of ten crime-related terms are considered. Table 4.2 shows the result for six articles and their status of acceptance or rejection based on their relatedness to the crime domain. The first three rows of the table are evident for true positive crime-related articles. The first row is an example of a trivial case where the article is selected based on the presence of the keyword “weapon” in both the headline and the corpus. The second and third rows show the selection of articles by applying LDA and using DBpedia as an external knowledge base. The probability score indicates the highest probability distribution score of an article towards a crime topic. The articles are selected as the probability score is more than the threshold and have the keywords whose entry is found in the DBpedia crime category. This helped to update the corpus with useful and unexpected keywords like “gang”, “gangster” and “rape” which were not available in the corpus previously. The fifth row shows the true negative article whose rejection as a crime article is trivial due to its probability score which is less than the threshold. The improper distribution of true negative articles towards a crime topic by the LDA is overcome by the use of DBpedia which is evident from the sixth row. Even though the article in the sixth row has the highest probability score, it is rejected due to the non-availability of any keyword related to the DBpedia crime category. The limitation of the proposed method is in filtering false-negative articles which is also evident from the fourth row. Although the article in the fourth row is confirmed to be crime-related by the probability score, due to the non-availability of any keyword re-

[6] https://www.kaggle.com/therohk/million-headlines

Table 4.2: Data gathering results for a sample of six articles

Initial corpus	Updation to the corpus	Sub-URL	Probability score	Keywords	DBpedia crime entry identified	Status
arrest, murder, kidnap, corrupt, theft, harassment, assault, blackmail, crime, weapon		https://www.indiatoday.in/crime/story/j-k-2-held-for-weapon-snatching-1475975-2019-03-12				✓
	gang, gangster	https://timesofindia.indiatimes.com/city/gurgaon/member-of-haryanas-lawrence-bishnoi-gang-held-in-hyderabad/articleshow/64495307.cms	0.9956219	Gang,Murder Gangster	http//dbpedia.org/resource/Category:Crimes http://dbpedia.org/resource/Category:Organized_crime_members_by_role	✓
	rape	https://www.indiatoday.in/crime/story/10-year-jail-for-raping-minor-in-mumbai-1477005-2019-03-13	0.7532135	Rape	http://dbpedia.org/resource/Category:Sex_crimes	✓
		https://timesofindia.indiatimes.com/city/mumbai/6-year-old-abducted-girl-found-dead-in-railway-toilet-in-navsari/articleshow/63462708.cms	0.9934596	Dead,Abduct	No DBpedia crime entry	✗
		https://www.deccanchronicle.com/lifestyle/health-and-wellbeing/300319/exercise-can-help-in-containing-arthritis.html	0.5414266			✗
		http://odishasuntimes.com/mahaprayan-ambulance-failure-in-odisha-body-carried-on-rickshaw-baby-delivered-in-auto/	0.9629939	Ambulance,Rickshaw, Delivered,Body, Baby,Auto,Odisha	No DBpedia crime entry	✗

lated to the DBpedia crime category, the article is categorized as a false negative. Such false negatives can be avoided by introducing more external knowledge bases along with DBpedia which will be considered in the future.

4.7.2 Evaluation of Information Extraction

The proposed algorithm's performance for entity and relation extraction from English news articles is compared to that of NLTK, a well-known tool for performing NLP tasks. To assess performance, NER tagged tuples obtained from NLTK and the proposed approach are compared in terms of Un-tagged and Erroneously tagged (UE-tagged) entities. The experiment is carried out on three types of sentences. The first category includes simple sentences with a single relation, while the second and third categories include sentences with two and three relations, respectively. The comparison result is shown in the table 4.3. From the table it is evident that, the accuracy of the proposed approach is significantly improved over the NLTK and is calculated as:

$$Accuracy = \frac{(Total\ Number\ of\ Entities) - (Number\ of\ UEtagged\ Entities)}{Total\ Number\ of\ Entities} \tag{4.5}$$

From the table, it is also evident that the proposed method has made a significant improvement for category-1 sentences when compared to the other two categories. As a

Table 4.3: Comparison of NLTK and proposed DBpedia-based approach

Category	NLTK				Proposed Approach			
	Total Number of Entities	Number of Wrongly tagged Entities	Number of Untagged Entities	Accuracy	Total Number of Entities	Number of Wrongly tagged Entities	Number of Untagged Entities	Accuracy
Category-1	60	12	5	71%	60	2	0	96%
Category-2	100	16	10	74%	100	6	15	79%
Category-3	150	25	18	71%	150	12	22	77%

result, the scope of our work is restricted to sentences with a single relationship. Similarly, when compared to un-tagged entities produced by NLTK, the proposed approach outperforms in tagging the entities with proper tags. This will have a significant impact on minimizing the redundancy in the knowledge base compared to the loss of information.

4.7.3 Evaluation of Semantic Merging

Experiments are carried out in semantic merging to find the most similar events using both word embedding and the knowledge base algorithm. The contextual similarity between the events is obtained by developing a word embedding model using a large collection of text as a training corpus. A pre-trained Word2Vec model of a news corpus provided by Google is used to accomplish this. The pre-trained model is loaded using Gensim, a Python framework for modeling vector space.

Here the experiment is conducted over the knowledge bases of different sizes that consist of events and entities extracted from multiple news articles. The effect of semantic merging is investigated separately for semantic similarity measures and a combination of semantic and contextual similarity measures and is shown in tables 4.4 and 4.5 respectively. The empirical study shows that the number of events and entities are reduced significantly using contextual similarity in combination with semantic similarity measure.

Although the table 4.5 is evident for the reduction in the size of the knowledge base after semantic merging, figure 4.10 shows the existence of redundancy even after semantic merging for varying sizes of the knowledge base. The presence of redundancy even after semantic merging is due to the wrong tags assigned to two or more semanti-

Table 4.4: Knowledge base before and after semantic merging- using only semantic similarity measure

Before Semantic Merging			After Semantic Merging		
Knowledge Base size (in KB)	Num of events	Num of entities	Knowledge Base size (in KB)	Num of events	Num of entities
49.8	6	29	49.8	6	29
137	22	105	91.8	20	90
283.3	56	126	166.9	56	100
410.5	85	221	269.7	82	206
1024	152	334	536.9	152	328

Table 4.5: Knowledge base before and after semantic merging- using both contextual and semantic similarity measure

Before Semantic Merging			After Semantic Merging		
Knowledge Base size (in KB)	Num of events	Num of entities	Knowledge Base size (in KB)	Num of events	Num of entities
49.8	6	29	40.4	4	29
137	22	105	88	16	90
283.3	56	126	160.1	51	95
410.5	85	221	261.7	78	198
1024	152	334	529.2	144	301

cally similar entities.

4.8 SUMMARY

This chapter presents a methodology to build a Bootstrapping Knowledge Base by extracting and integrating crime entities and their relations from text and image data of online newspapers in the English language with an emphasis on minimizing redundancy and loss of information. The proposed algorithm to extract the entities uses DBpedia as background knowledge. Because of the common source for extraction, the generation of erroneously tagged entities and hence the redundancy is avoided. Similarly,

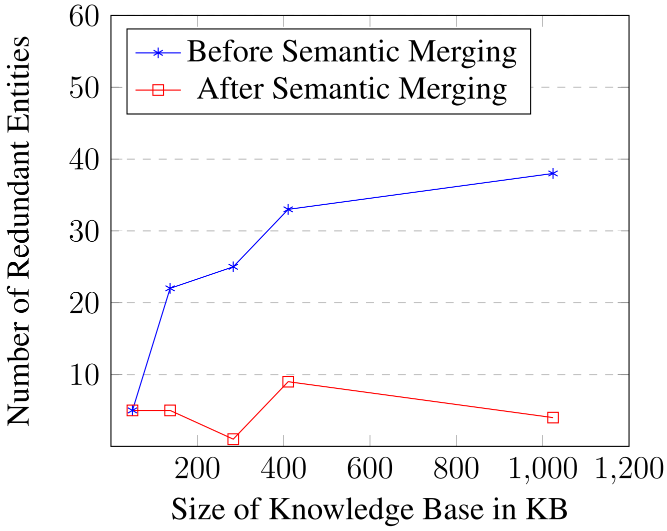


Figure 4.10: Redundant entities before and after semantic merging

when compared to un-tagged entities produced by NLTK, the proposed approach outperforms in tagging the entities with proper tags. The empirical results show that the proposed approach for information extraction achieves the highest accuracy of 96% for sentences with a single relation compared to the entities extracted from NLTK. However, the generation of un-tagged entities and hence the loss of information still exist if an entity can not link to any DBpedia property. The chapter also explores the use of both contextual as-well-as semantic similarity measures for finding the similarity between the entities. Empirical results show that redundancy is minimized to a greater extent by using both similarity measures.

CHAPTER 5

A KNOWLEDGE BASE COMPLETION FRAMEWORK

5.1 INTRODUCTION

Extracting the facts namely entities and relations from unstructured sources is an essential step in any knowledge base construction. At the same time, it is also necessary to ensure the completeness of the knowledge base by incrementally extracting the new facts from various sources. To date, knowledge base completeness has been investigated as a problem of knowledge refinement, in which missing facts are inferred from information already in the knowledge base (Pezeshkpour et al. (2020)). Facts that are missing while extracting data from multilingual sources, on the other hand, are ignored.

Given a set of bi-lingual news articles from resource-rich Source Language (SL) like English and resource deficit Target Language (TL) like Hindi respectively. This chapter aimed at developing a framework for knowledge base completion to extract the facts from TL news articles so that the KB created using SL news articles is enriched with new facts available in TL news articles. This is to be achieved by exploiting the redundancies available from SL and TL news articles and without using the language-specific tools for TL news articles.

The primary contributions of this chapter include:

- Proposed an algorithm for grouping the related news articles in a bilingual corpus.
- Proposed a bootstrapping-based method to extract the facts from low-resourced

language Hindi news articles using facts extracted from high-resourced English language news articles without using language-specific tools for Hindi.

5.2 SYSTEM ARCHITECTURE

The detailed architecture of the proposed framework is shown in figure 5.1. The proposed framework performs bootstrapping at multiple levels to extract the new facts from Hindi news articles using the triples extracted from the related English news articles. The goal of the chapter is to develop a generic framework for knowledge base completion so that a bootstrapping knowledge base of crime-related facts extracted from online new articles in the English language as mentioned in chapter 4 is supplemented with facts extracted from low-resourced Indian language Hindi news articles. To accomplish this, a clustering algorithm is proposed that, unlike the previous Bag of Words representation, examines the redundancy among the bi-lingual collection of news articles by modeling clusters with knowledge base facts. Facts extracted from English language articles are bootstrapped to extract facts from equivalent Hindi language articles from each cluster. Accordingly, the proposed framework is broken into two phases namely, clustering and bootstrapping-based extraction and are discussed in the following sections.

5.3 CLUSTERING

Initially, the crime-related Hindi articles are selected by applying topic modeling and knowledge base aided data acquisition method proposed in 4.3 over the headlines translated to English. The redundancies among the articles are exploited by identifying the comparable articles. A set of bi-lingual collections of articles is said to be comparable if they are related either topically or sententially. Topically related articles are contextually similar articles that discuss the same topic and are said to be semantically related. Whereas, sententially related articles are almost bi-lingual translations of each other and are said to be semantically similar. Hence topically related articles are enriched with more new information compared to sententially related articles. The proposed work identifies the comparable articles using the semantic merging procedure mentioned in 4.5. As the news articles are published daily, the articles are considered as data

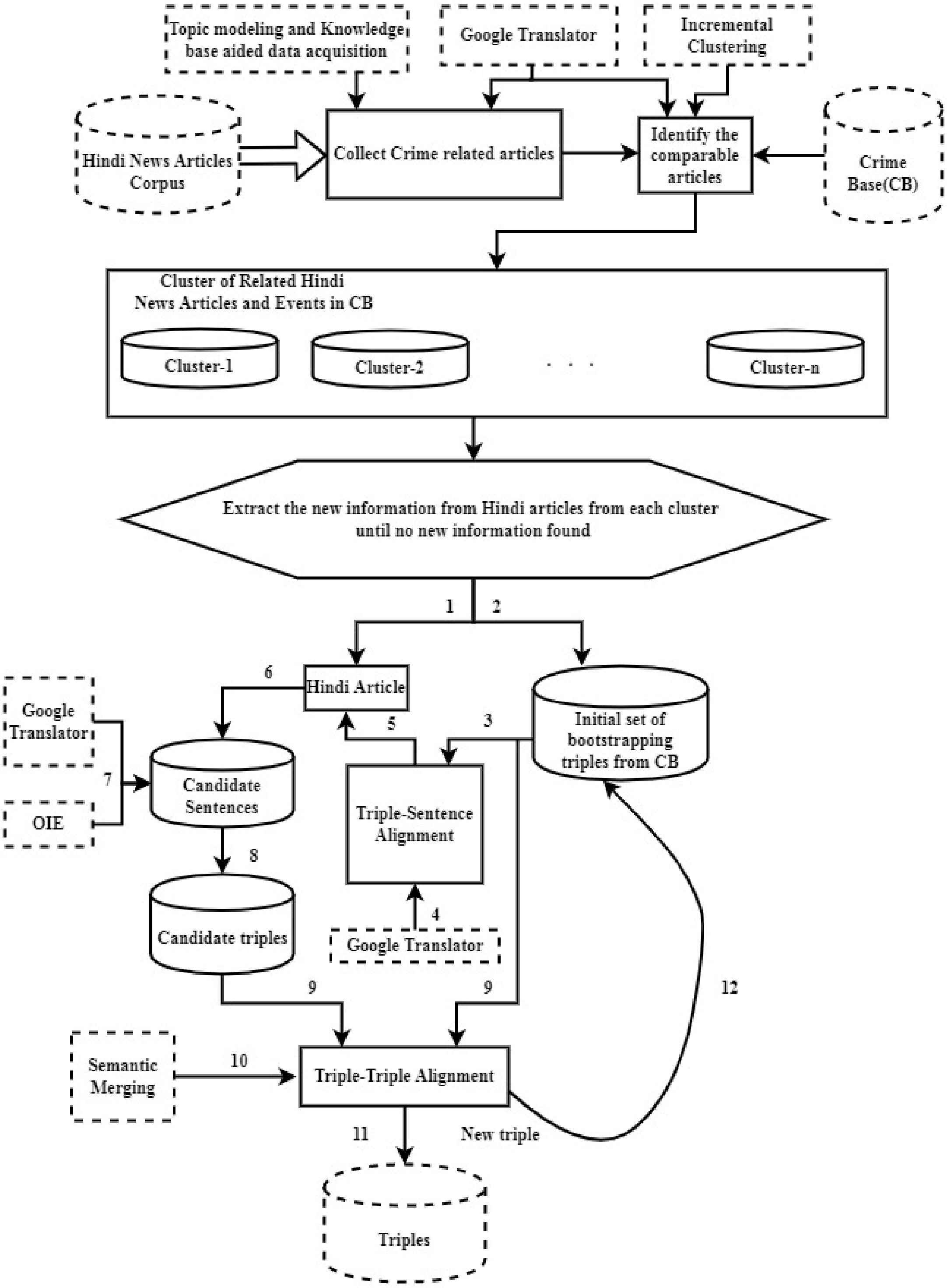

Topic modeling and Knowledge base aided data acquisition
Google Translator
Incremental Clustering
Hindi News Articles Corpus
Collect Crime related articles
Identify the comparable articles
Crime Base(CB)
Cluster of Related Hindi News Articles and Events in CB
Cluster-1
Cluster-2
. . .
Cluster-n
Extract the new information from Hindi articles from each cluster until no new information found
1
2
6
Hindi Article
Google Translator
5
3
Initial set of bootstrapping triples from CB
7
Candidate Sentences
Triple-Sentence Alignment
OIE
8
Candidate triples
4
Google Translator
9
9
12
Semantic Merging
10
Triple-Triple Alignment
11
New triple
Triples

Figure 5.1: *Detailed architecture of the proposed framework*

streams and an incremental nearest neighborhood algorithm for clustering data streams is adopted (Louhi et al. (2016)). Here, the clustering algorithm is modified to identify the sententially and topically related articles by finding sentential and topical neighbors and is named as Sentential-Topical-Nearest-Neighborhood (STNN) algorithm which is described in Algorithm 4.

The major difficulty in clustering articles is in semantically representing the articles so that, clusters of better quality can be formed. Due to the availability of a large number of terms as document features, the Bag of Words way of representing the documents does not capture the semantics hidden in the sentences. To improve the semantics, the articles are represented as KB facts in the form of triples extracted over the headlines. Accordingly, the headlines from Hindi news articles are translated to English, and facts from the translated headlines are extracted using the method proposed in 4.4. When a stream of facts from English and Hindi news articles comes in, we divide them into various windows based on their date of publication. Now, events in the first window are clustered using neighborhood-based clustering. The similarity between each of the elements in the first window is calculated using contextual as-well-as semantic similarity measures. The significance of using both the similarity measures is empirically proved and can be found in 4.5. Two elements are considered to be topically neighbors if their contextual similarity is greater than a threshold value. Such neighbors are also checked for their semantic similarity. If the semantic similarity is greater than their contextual similarity score, they form a separate cluster and will be added to the set of sententially similar clusters. Otherwise, they will be added to the set of topically similar clusters. If the contextual similarity score for any two elements is less than the threshold, the elements are independent and form two separate clusters. To represent a cluster, we find the medoid of each cluster, where the medoid is an element that has the maximum similarity with all other elements in the cluster. This limits further comparison between the medoids rather than with all the elements in the cluster. A similar method is followed to find the clusters for other windows. For each new cluster, we find from the former clusters the most similar cluster to them by calculating the similarity of the medoid event of the former clusters and the medoid of the new cluster. Based on their similarity, two

Algorithm 4: STNN Algorithm

```
Input: $E = \{F_{E_1}, F_{E_2}, ..., F_{E_m}\}$:Set of $m$ crime facts extracted from English
       news articles and $H_H = \{H_1, H_2, ..., H_n\}$:Set of $n$ headlines extracted
       from Hindi language news articles
Output: Set of $n_s$ sententially similar clusters $C_s = \{C_1, C_2, .., C_{n_s}\}$, Set of $n_t$
        topically similar clusters $C_t = \{C_1, C_2, .., C_{n_t}\}$
Translate the headlines in Hindi to English and the set of translated headlines
  be $H_{H_T} = \{H_{t_1}, H_{t_2}, ..., H_{t_n}\}$
Extract the facts from $H_{H_T}$ and let $H = \{F_{H_1}, F_{H_2}, ..., F_{H_k}\}$ be a set of $k$ facts
  related to Hindi headlines
Divide the events from $E$ and $H$ into multiple windows $W = \{w_1, w_2, ...\}$
  where $w_i \subseteq E \cup H$ indicates facts extracted from the articles published during
  $i^{th}$ date
Find the neighbors and hence clusters for $w_1$ as follows:
Calculate contextual similarity $CS$ between each new couple of elements $F_{E_i}$
  and $F_{H_j}$
if $CS >$a threshold $t_c$ then
    Calculate semantic similarity SS between each couple
    if $SS >$ a threshold $t_s$ then
        The elements are sententially neighbors. Each set of neighbors
          represent a cluster and will be added to $C_s$
    else
        The elements are topically neighbors. Each set of neighbors represent a
          cluster and will be added to $C_t$
else
    Add the elements to $C_t$ as new clusters
Find medoid of each cluster where, medoid is the element which has the
  maximum similarity with all the elements in the cluster
Similarly find the neighbors and hence the clusters for the subsequent windows
Calculate new clusters medoids
Calculate the similarity between new medoids and medoids of old clusters
if found a pair of contextually or semantically similar medoids then
    Merge the clusers
    Update medoid
    Add the merged cluster to the appropriate set
else
    Retain the clusters as it is
```

clusters are merged and the medoid will be updated.

5.4 EXTRACTION

In this work, we propose a method to identify and extract the new facts from a target language news article like Hindi using the facts extracted from related English news articles. This is achieved by bootstrapping the triples extracted from English news articles to identify the presence of related triples from comparable Hindi news articles. The proposed extraction method constitutes two steps namely:

1. Candidate sentence identification
2. New triple generation

Each of the steps is explained in the following subsections.

5.4.1 Candidate sentence identification

From each cluster, the events related to English news articles are selected as an initial set of bootstrapping triples. Each of these triples is translated to the target language using Google translator API and used to query the Hindi articles to identify a set of sentences that are enriched with new facts and are called candidate sentences. Given a set of bootstrapped triples from English articles $B_E = \{t_{E_1}, t_{E_2}, ..., t_{E_n}\}$, a set of candidate sentences from Hindi articles $S = \{s_1, s_2, ..., s_m\}$ are obtained by aligning the sentences with the triples. Formally, a sentence s_i is said to be aligned with t_{E_j}, if an element e_k belongs to t_{E_j} is a substring of s_i. Finally, a sentence that constitutes the un-aligned part in it is selected as the candidate sentence. Otherwise it is considered as similar to t_{E_j}. Figure 5.2 illustrates the generation of candidate sentences with an example.

5.4.2 New triple generation

Once the candidate sentences are extracted, new triples are obtained in three steps namely:

1. Candidate sentence translation.

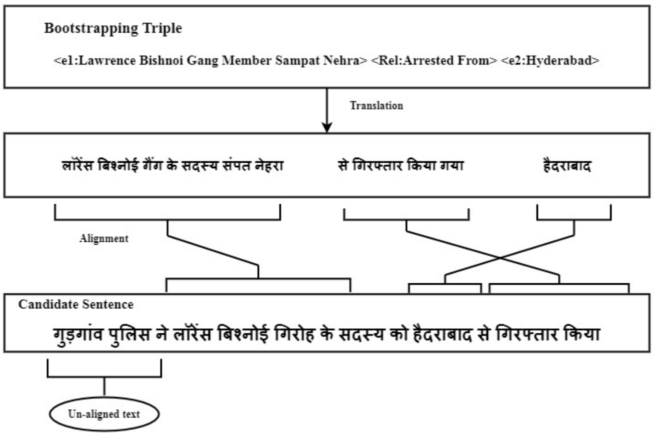

Bootstrapping Triple
<e1:Lawrence Bishnoi Gang Member Sampat Nehra> <Rel:Arrested From> <e2:Hyderabad>
Translation
लॉरेंस बिश्नोई गैंग के सदस्य संपत नेहरा
से गिरफ्तार किया गया
हैदराबाद
Alignment
Candidate Sentence
गुड़गांव पुलिस ने लॉरेंस बिश्नोई गिरोह के सदस्य को हैदराबाद से गिरफ्तार किया
Un-aligned text

Figure 5.2: *Candidate sentence generation*

2. Triple/s extraction.

3. Projection of triple.

Initially, candidate sentences are translated to the English language using Google API translator, and triples from each sentence are extracted using the method proposed in 4.4. Triples so extracted from a candidate sentence are projected against the bootstrapped triple to identify the new triples as shown in figure 5.3 with the continuation of the example considered in figure 5.2.

5.5 EXPERIMENTAL RESULTS

This work considers a prominent newspaper *Hindustan* for Hindi News articles which has articles available online. The corpus includes the data collected from Jan 2018 to Jun 2018. The following sections describe the experimental evaluation results for clustering and extraction.

5.5.1 Evaluation of Clustering Algorithm

The proposed algorithm for clustering is evaluated in two phases namely, bi-lingual and mono-lingual evaluation. In the first phase, the algorithm is evaluated for English and Hindi articles and in the second phase, the algorithm is evaluated for English articles. Due to the lack of algorithms for clustering multi-lingual articles, a baseline algorithm i.e. incremental nearest neighborhood algorithm without using background KB and

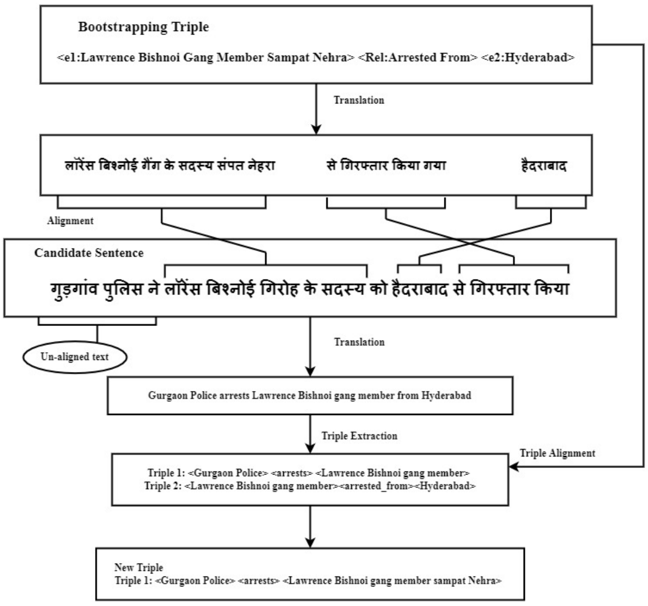

Bootstrapping Triple
<e1:Lawrence Bishnoi Gang Member Sampat Nehra> <Rel:Arrested From> <e2:Hyderabad>
Translation
लॉरेंस बिश्नोई गैंग के सदस्य संपत नेहरा
से गिरफ्तार किया गया
हैदराबाद
Alignment
Candidate Sentence
गुड़गांव पुलिस ने लॉरेंस बिश्नोई गिरोह के सदस्य को हैदराबाद से गिरफ्तार किया
Un-aligned text
Translation
Gurgaon Police arrests Lawrence Bishnoi gang member from Hyderabad
Triple Extraction
Triple Alignment
Triple 1: <Gurgaon Police> <arrests> <Lawrence Bishnoi gang member>
Triple 2: <Lawrence Bishnoi gang member><arrested_from><Hyderabad>
New Triple
Triple 1: <Gurgaon Police> <arrests> <Lawrence Bishnoi gang member sampat Nehra>

Figure 5.3: *New triple generation*

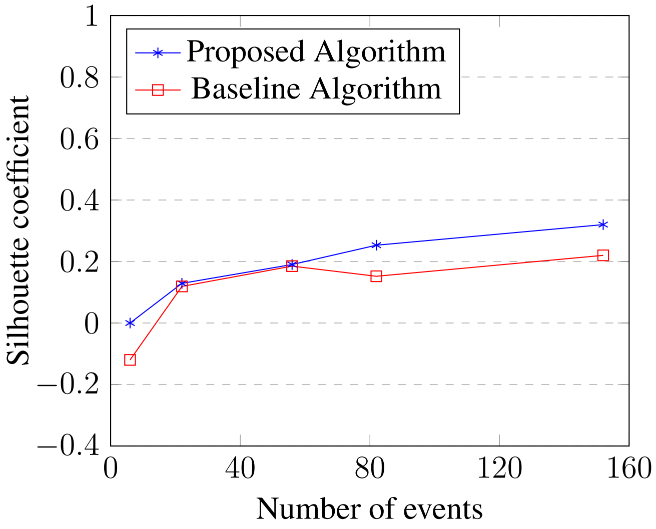


Figure 5.4: Bi-lingual evaluation of proposed clustering Algorithm: Silhouette coefficient for varying number of events

considering only the headlines from English and Hindi news articles is implemented.

5.5.1.1 Bi-lingual Evaluation

The proposed algorithm is compared with the baseline algorithm in terms of the quality of clusters formed and the time taken for clustering. The clustering quality is determined using a Silhouette coefficient (Rousseeuw (1987)). This is a well-known measure of internal evaluation for evaluating clusters without pre-determined labels. It measures how similar an object is to its cluster compared to other clusters. The Silhouette coefficient for i^{th} event is calculated as follows:

$$s_i = \frac{a_i \; - \; b_i}{max \; (\; a_i \; , \; b_i)} \tag{5.1}$$

where a_i is the average similarity of the i^{th} event with all the other events in its cluster. Then for all the other clusters to which i^{th} event does not belong, we calculate the average similarity of i^{th} event to all the events in these clusters and b_i is the maximum of all these values. Figure 5.4 shows the silhouette coefficient obtained for proposed and baseline algorithms for varying numbers of events. We can see that the proposed algorithm achieved a larger value of silhouette coefficient as the event size increases and hence produced a better quality of clusters. However, the Silhouette coefficient value is not very close to 1 because of many individual clusters obtained during the clustering process. These events are those which do not have a similarity with any other crime events. If we do not consider the individual clusters, then we get an average value of 0.63 and 0.45 as silhouette coefficients for proposed and baseline algorithms respectively.

We also evaluated the quality of clusters in terms of the number of related events obtained for a given keyword. Table 5.1 displays some of the input keywords used for finding clusters of related events over a cluster. From the table, it is clear that, due to the higher quality of clusters formed by the proposed algorithm, the number of related events associated with a given keyword is also significantly high.

Table 5.2 shows the clustering time taken by the proposed and baseline algorithms. From the table, it can be observed that the proposed algorithm takes more time compared to the baseline approach. This is due to an additional cost incurred from machine

Table 5.1: Number of related events for keywords before and after clustering

Keyword	Number of related events (Baseline algorithm)	Number of related events (Proposed algorithm)
Kanpur	1	3
Navsari	2	2
Venugopal	3	4
Malad	1	3
Mumbai	6	14
Bandipora	2	2
CRPF	7	9
Railway_Act	1	1
Abhijit Mukherjee	1	3
Kaluram	6	6
Congress	15	26

translation and extraction of two or more triples from a single headline. However, more semantics hidden in triples compared to raw sentences produces clusters with high quality, and hence the time complexity is compromised over the cluster quality.

5.5.1.2 Mono-lingual Evaluation

Here the proposed work is evaluated by considering only the English news articles and comparing the results with two recently proposed works (Bisandu et al. (2018)) and

Table 5.2: Bi-lingual evaluation of proposed clustering algorithm: Time taken for clustering

Features (Bag of words in terms of Headlines)	Clustering time for Baseline algorithm(in sec)	Features (Events in terms of Triples)	Clustering time for proposed algorithm (in sec)
100	92	270	98
200	194	423	222
300	298	610	343
400	372	908	402
500	536	1022	582

Table 5.3: Mono-lingual evaluation of proposed clustering algorithm

Methods	Datasets	Accuracy	Purity
Bisandu et al. (2018)	Reuters	0.3950	0.9418
	20Newsgroups	0.3801	0.9200
Sohangir and Wang (2017)	Reuters	0.2320	0.5769
	20Newsgroups	0.1659	0.4234
Proposed Algorithm	Reuters	0.5210	0.6200
	20Newsgroups	0.4832	0.7398

(Sohangir and Wang (2017)) as a baseline. Evaluation in these two works is done using Reuters and 20Newsgroup datasets. The details about the datasets can be found in (Sohangir and Wang (2017)). (Sohangir and Wang (2017)) uses a K-means clustering algorithm with improved square root similarity measure. As an improvement to this, (Bisandu et al. (2018)) used N-grams representation along with K-means clustering algorithm and improved square root similarity measure. The proposed algorithm is different from the baseline works by using semantically rich triples representation and a similarity measure using both contextual and semantic similarity measures proposed in 4.5. Here, the experiment is conducted using two thousand samples each from Reuters and 20Newsgroup datasets over five newsgroups. The triples are extracted from each sample using the method proposed in 4.4. To speed up the execution, a parallel version of the proposed clustering algorithm is implemented using Message Passing Interface (MPI). The triples are processed in parallel to identify the clusters.

Table 5.3 shows the evaluation results for the proposed and the baseline approaches. The same performance metrics as mentioned and defined in (Bisandu et al. (2018)) i.e. accuracy and purity are used here for evaluation. From the table, it is clear that the proposed clustering algorithm performs better than baseline methods in terms of accuracy. However, due to the generation of more individual clusters i.e. clusters with a single element, the purity of the proposed algorithm is less compared to (Bisandu et al. (2018)).

5.5.2 Evaluation of Extraction

To evaluate the results for the proposed KBC approach, a Machine Translation (MT)-based system is implemented which is considered as a gold standard. The gold standard

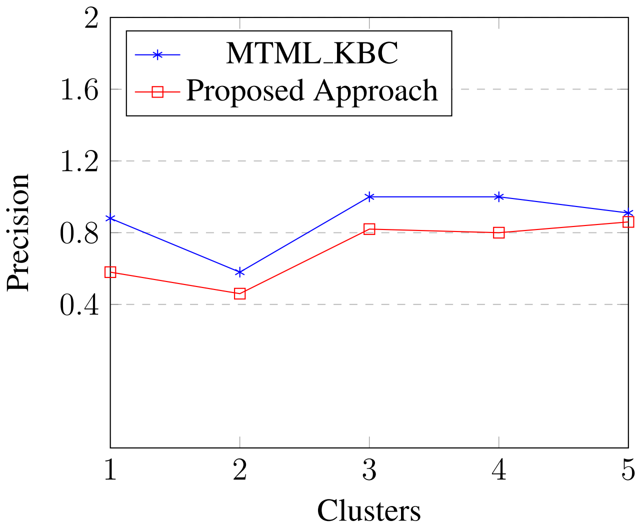


Figure 5.5: Precision for five clusters

system reduces the problem to mono-lingual information extraction and integration by translating the entire articles in the target language into English. Then the facts are extracted from translated articles and are semantically merged with facts extracted from English news articles using the methods for IE and semantic merging proposed in chapter 4. Hence the gold standard system is named as Machine Translation based Monolingual Knowledge Base Completion (MTML_KBC). The quality of the proposed KBC approach is measured using the standard evaluation metrics precision and recall. Precision is calculated as the ratio of the number of valid new facts extracted to the total number of new facts extracted. The recall is calculated as the ratio of the number of valid new facts extracted to the total number of valid new facts available.

Table 5.4 shows the results recorded for five different clusters. Figures 5.5 and 5.6 shows the performance of gold standard (MTML_KBC) and proposed approach in terms of precision and recall respectively. From the figures, it is clear that the proposed approach achieves a better recall compared to precision. This is evident from the fact that the total number of new facts extracted by the proposed approach is more due to improper projection of bootstrapping triples with the triples extracted from candidate sentences.

Table 5.4: Comparison of MTML_KBC and proposed approach

Clusters	MTML_KBC			Proposed Approach		
	Total New facts Extracted	Number of new facts available	Number of valid new facts extracted	Total New facts Extracted	Number of new facts available	Number of valid new facts extracted
Cluster-1(52 facts+13 Hindi articles)	9	10	8	12	10	7
Cluster-2(83 facts+08 Hindi articles)	12	9	7	15	9	7
Cluster-3(75 facts+18 Hindi articles)	14	15	14	17	15	14
Cluster-4(92 facts+11 Hindi articles)	18	20	18	21	20	17
Cluster-5(88 facts+14 Hindi articles)	23	25	21	23	25	20

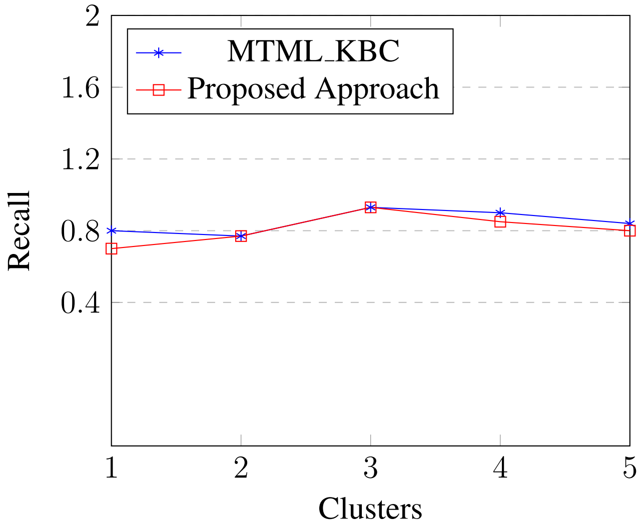


Figure 5.6: Recall for five clusters

5.6 SUMMARY

This chapter proposed a clustering and bootstrapping-based generic framework for knowledge base completion. Using the framework, any knowledge base created with the facts extracted from English news articles can be enriched with new facts available in low-resourced language articles without using language-specific tools. Here the experiment is conducted using the low-resourced Indian language Hindi news articles. The redundancies that exist among the bi-lingual collection of articles are exploited by grouping the articles that are topically or sententially similar using the nearest neighborhood clustering. The proposed clustering algorithm makes use of knowledge base facts in terms of triples to represent the articles against the traditional Bag of Words representation, as the triples capture the high semantics. Empirical results show that the proposed algorithm takes more time compared to the baseline approach due to the extraction of two or more triples from a single headline. However, the clusters of high quality with an average value of 0.63 as silhouette coefficients is obtained for proposed algorithms using bi-lingual facts. The proposed algorithm is also evaluated for mono-lingual facts by comparing the results with two recently proposed works over Reuters and 20Newsgroup datasets. The proposed algorithm achieved the highest accuracy of 0.5210 and 0.4832 for Reuters and 20Newsgroup datasets respectively. However, due to the generation of more individual clusters i.e. clusters with a single element, the purity of the proposed algorithm is less compared to an existing work with N-grams representation along with K-means clustering with improved square root similarity measure. From each group of related articles, the facts related to English news articles are bootstrapped to extract the facts from Hindi news articles using Google translator API. This way of using the high-resource language facts as bootstrapping triples helps to extract the facts from articles related to the languages for which language processing tools like POS tags are neither available nor accurate. Experimental results for extraction show that using the framework a better recall of 93% is achieved in identifying the new facts compared to precision. This is evident from the fact that the total number of new facts extracted by the proposed approach is more due to improper projection of bootstrapping triples with the triples extracted from candidate sentences. We ran 1000 triples through Googletrans, a python library for Google Translator to check the accuracy of google translator. The results are manually verified, and the translator's accuracy is 98.3 percent, except for a few proper nouns, such as "Gurgaon", where the translation did not match.

CHAPTER 6

TRIPLE BASED FAKE NEWS CLASSIFICATION

6.1 INTRODUCTION

The advancement of information technology has resulted in a massive increase in the number of people reading online news stories. At the same time, the generation and circulation of fake news are on the rise, potentially putting people at risk. According to Gartner, “by 2022, the majority of individuals in advanced economies will consume more incorrect information than correct information”, (Thota et al. (2018)). Fake news can be any rumor or false information spread to deceive readers, harm a company’s reputation, or profit from sensationalism. Fake news must be detected automatically since it undermines an event’s legitimacy. For instance, “there is a huge drop in attendance for a certain vaccination camp as there was a rumor stating that the vaccination camp is a population control camp” is a serious threat to society and should be detected as soon as possible.

A lot of work has gone into developing methods and tools for successfully extracting facts from internet news items in the realm of knowledge base creation. Validation of facts, on the other hand, has received the least attention in the available literature. Models to measure the semantic correctness of facts from the knowledge graph have been created recently, such as (Jia et al. (2019)). For example, these models identify whether a relationship exists between two entities, and such facts are inferred from the knowledge graph’s existing facts. These models, on the other hand, are unable to detect any fraudulent facts added to the knowledge base. As a result, before facts are added to the knowledge base, they must be checked for truth or falsity.

Checking the source of the news to see if it is genuine or not is a common method of determining whether it is true or not. However, tracing the origin is an NP-Hard task. In recent years, the subject of detecting fake news has been explored as a classification problem, where a news piece is classed as either true or false. Current efforts to detect fake news rely on the development of sophisticated machine learning and deep learning models based on a variety of news datasets. For instance, in (Agarwal et al. (2020)) authors proposed a model using neural networks for fake news detection. Similarly, recent works like (Ghosh and Shah (2018); Gupta and Meel (2021); Reddy et al. (2020); Shu et al. (2019, 2017); Thota et al. (2018); Zhang et al. (2020); Zhou et al. (2019)) proposed their model for detecting fake news using different variants of the machine and deep learning. Recently, (Priyanga et al. (2021)) explored the use of the word and sentence embeddings in detecting fake news. Similarly, (Albahr and Albahar (2020); Khan et al. (2021a)) empirically studied the benchmark models for detecting fake news using various machine learning models. These models, on the other hand, are not intended to train knowledge base facts and are unsuitable for classifying a given triple as fake or true. Furthermore, extant works embed the features using either frequency or prediction-based word embedding models. Frequency-based word embeddings, such as TF-IDF, generate document vectors that identify the importance of words to a document by generating vectors of dimension $D\ X\ N$, where D is the number of documents and N is the number of terms in the corpus. However, these methods generate vectors with a very high dimension and consume a lot of memory. This is overcome by the word vectors generated by the prediction-based word embedding models. These models, on the other hand, extract word features based on their co-occurrence with other words while ignoring document-level features. Thus, it is critical to investigate the accuracy of models trained over features extracted using both word embeddings. As a result, using both word embedding models, this chapter considers the implementation of a deep learning-based multi-layer perceptron classifier to empirically test the effectiveness of employing triples for fake news identification. Despite the fact that a classifier can be trained using only triples without Named Entity Tags (NETs), NETs are an additional feature, and thus the variation in accuracy of classifiers with and without NETs is also investigated in this work.

The primary contributions of this chapter include:

- Proposed the use of Multi-layer Perceptron with state-of-the-art Word2Vec (Church (2017)) and GloVe (Pennington et al. (2014)) word embeddings along with traditional TF-IDF and count vectorizers to classify the facts extracted from news articles as true or fake. A data modeling approach is proposed that investigates feature extraction using a combination of frequency and prediction-based word embedding models, allowing for the consideration of both document-level and word-level features. This also aids in the reduction of high-dimensional vectors generated by both frequency and prediction-based word embedding models to low-dimensional vectors.

- Investigated a machine learning-based baseline model for comparison by implementing six machine learning-based algorithms trained over the triples.

6.2 SYSTEM ARCHITECTURE

Overview of the proposed method for fake news detection using triples is shown in Figure 6.1. It consists of two parts namely, extraction of triples and training the models over the triples. Former is performed using the algorithm proposed in 4.4. This chapter covers methodology followed for later i.e. to train the machine and deep learning models using triples, and experimental results for the same. A trained model can be used to classify a triple as fake or true.

6.3 MODEL CONSTRUCTION

From the perspective of knowledge base creation, the goal of this chapter is to create a deep learning-based classifier for fake news detection. The model should be able to determine whether a triple taken from a news article is fake or true. This is accomplished in three stages namely, data modeling, baseline model investigation, and application of the suggested Multi-layer Perceptron model.

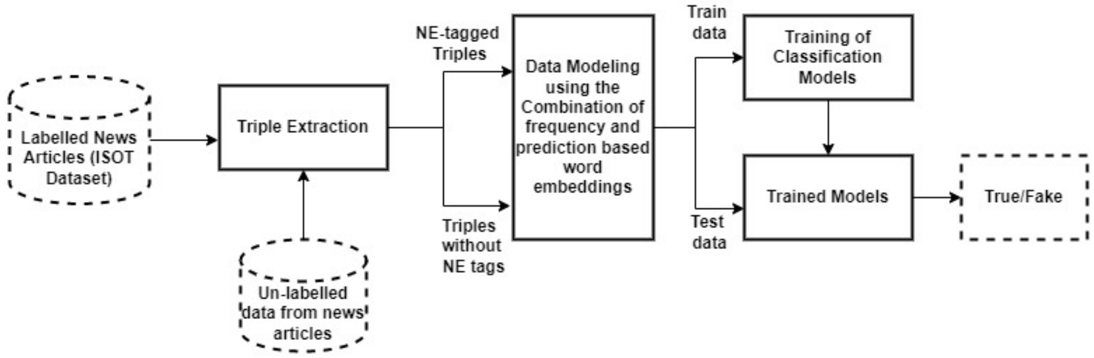

Labelled News Articles (ISOT Dataset)
Triple Extraction
Un-labelled data from news articles
NE-tagged Triples
Triples without NE tags
Data Modeling using the Combination of frequency and prediction based word embeddings
Train data
Test data
Training of Classification Models
Trained Models
True/Fake

Figure 6.1: Overview of the proposed work

6.3.1 Data modeling

Data modeling involves the generation of numerical representations for each triple in the corpus using word embedding models. Since textual data does not work well with machine learning and deep learning models, word embedding helps in lowering the dimensions of the data by converting them to integer or real-valued numerical representations. These numerical representations tend to capture their semantic meaning and try to understand the context in which they were used. Word embedding can be either frequency-based or prediction-based. In this work, two frequency-based and prediction-based word embedding techniques are used. Frequency-based word embedding is achieved using TF-IDF and count vectorizer. Whereas, prediction-based word embedding is performed using GloVe and Word2Vec and are explained in brief as follows:

- **TF-IDF:** Term Frequency-Inverse Document Frequency indicates how important a triple is for an article to the entire corpus. TF measures the frequency of a triple in an article and is calculated as:

$$TF(t,d) = \frac{t}{T} \tag{6.1}$$

Where t is the number of times a triple occurs in an article and T is the total number of triples in the article. IDF measures the uniqueness of a triple across the corpus and is calculated as:

$$IDF = log(\frac{N}{n}) \tag{6.2}$$

Where, n is the number of articles a triple is present in and N is the total number of articles.

- **Count Vectorizer:** Count vectorizer en-codes a given triple into a vector based on the frequency of each triple that occurs in the entire article. It vectorizes an article by creating a matrix in which each unique triple is represented by a column of the matrix, and each sample from the article is a row in the matrix. The value of each cell is nothing but the count of the triple in that particular sample.

- **GloVe:** GloVe termed as Global Vectors is a distributed model for word representation. Vector representation for words is obtained using an unsupervised learning algorithm. This is achieved by mapping triples into a meaningful space where the distance between triples is related to semantic similarity.

- **Word2Vec:** Word2Vec is one of the most popular techniques to learn word embeddings using a shallow neural network. It is designed to predict the word given a context window and vice versa.

Proposed Data Modeling Approach: The proposed data modeling approach vectorizes the triples by combining frequency and prediction-based word embedding models. The concept underlying the proposed approach is depicted in the figure 6.2. Frequency-based models, such as TF-IDF, generate document vectors of the order $D\ X\ N$ for a corpus with D documents and N terms. For a large corpus, these models generate vectors with extremely high dimensions, which do not fit into memory. Prediction-based models, on the other hand, generate word vectors of the order $N\ X\ P$, where P is much smaller than N and D, whose value is fixed while training the model, and thus reduce vector dimensionality. These models, however, do not take into account the document's features. To leverage the power of both models, this work vectorizes triples by taking the product of the vectors generated by both models. The final vector generated will be of the dimension DXP, embedding both document and word-level features while also reducing the dimensionality of the vector generated by both frequency and prediction-based word embeddings.

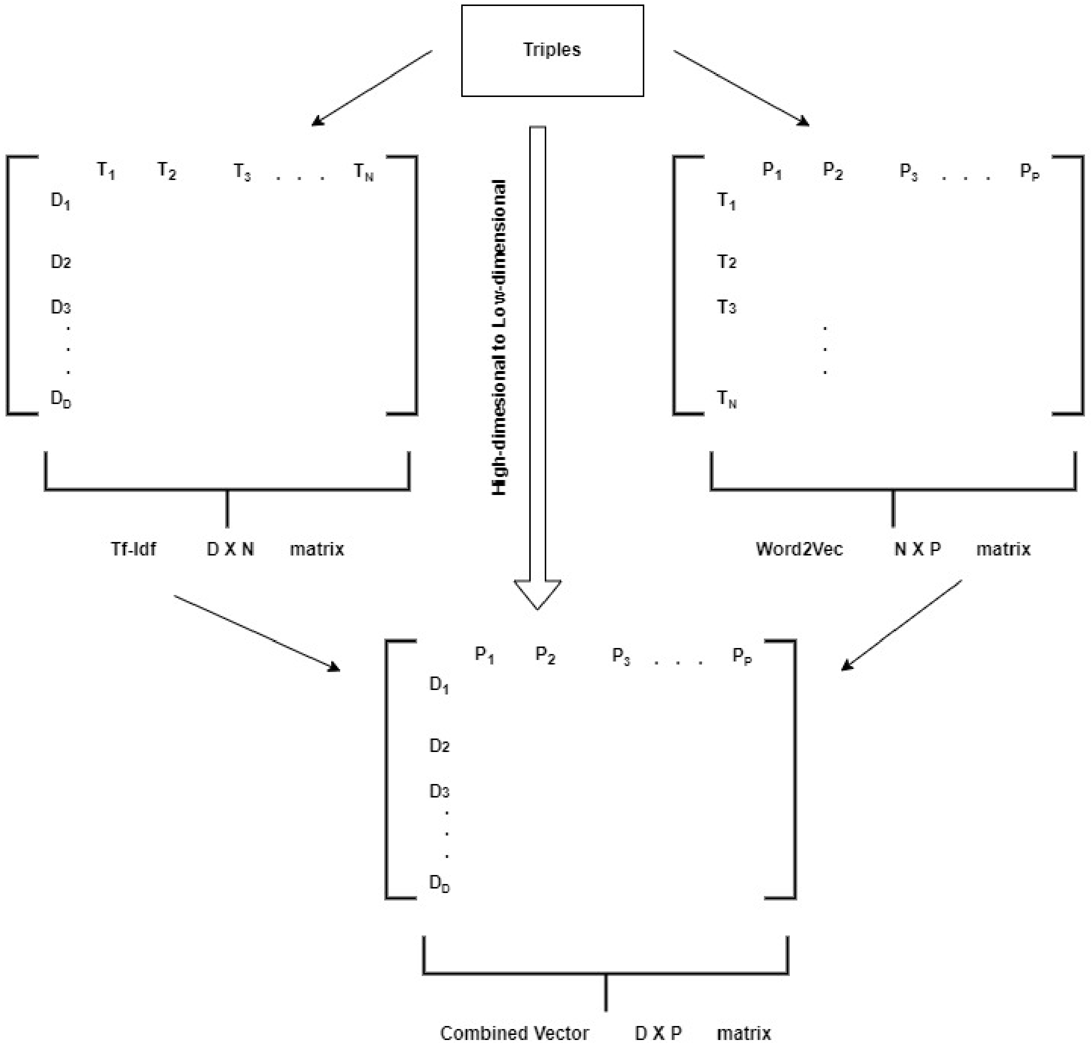


Figure 6.2: Proposed data modeling approach

6.3.2 Investigation of baseline model

Since it is a first attempt to train a deep learning model using triples for fake news classification, it is of utmost necessity to identify a baseline model for comparing the performance of the proposed model. This is achieved by training six machine learning classifiers using triples. Based on the previous works (Manzoor et al. (2019)) and (Ahmad et al. (2020)) for fake news detection using machine learning algorithms, six machine learning algorithms namely Multinomial Naive Base (MNB), Passive-Aggressive (PA), Support Vector Machine(SVM), K-Nearest Neighbors(KNN), Logistic Regression(LR) and Random Forest(RF) classifiers are trained using triples in this work. The classifier that produces the high accuracy is used as the baseline for comparing the results obtained using the proposed deep learning-based Multi-layer Perceptron. Each of the machine learning classifiers is explained in brief as follows:

- **Naive Bayes (NB) classifier:** NB calculates the conditional probability of a specific triple t, given relative frequency of t in articles belonging to a particular class. There are three types of NB models namely Gaussian, Multinomial and Binomial. In this work, we used the multinomial model as it is used for text classification problems, and prediction is based on the frequency of words. A sample code used for the implementation is shown below and the same will be used for other classifiers with a small change in a few parameters related to the respective classifiers.

```
clf = MultinomialNB()
clf.fit(tfidf_train, y_train)
pred = clf.predict(tfidf_test)
score = metrics.accuracy_score(y_test, pred)
print(”accuracy: %0.3f” % score)
clf.fit(count_train, y_train)
pred = clf.predict(count_test)
score = metrics.accuracy_score(y_test, pred)
print(”accuracy: %0.3f” % score)
```

- **Passive-Aggressive classifier:** The passive-aggressive algorithms are a family of algorithms for large-scale learning. They are similar to the Perceptron in that they do not require a learning rate, but they include a regularization parameter C. The ‘squarehinge’ loss function is used for classification. The classifier remains passive for a correct classification outcome, and it turns aggressive if it is an incorrect classification by updating and adjusting.

- **Support Vector Machine (SVM):** SVMs have supervised learning models with associated learning algorithms that analyze data for classification and regression analysis. Given a set of training examples for fake and true news, SVM builds a model that assigns one of the categories to new examples and hence becomes a non-probabilistic binary linear classifier. SVM maps training examples to points in space to maximize the width of the gap between the two categories. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall.

- **K-Nearest Neighbors classifier:** The K-Nearest Neighbors (KNN) algorithm is a supervised machine learning algorithm that is simple and easy to implement. This is used to solve both classification and regression problems.

- **Logistic Regression classifier:** Logistic Regression is a supervised learning classification algorithm. It is a predictive analysis algorithm based on the concept of probability. Logistic Regression can be called a Linear Regression model but the Logistic Regression uses a more complex cost function, ‘Sigmoid function’ or also known as the ‘logistic function’ instead of a linear function.

- **Random Forest classifiers:** Random forest is an ensemble-based algorithm, consists of a large number of individual decision trees as its name indicates. A class will be predicted by each tree in the forest and the class with the most votes becomes the model’s prediction.

6.3.3 Implementation of Multi-layer Perceptron Model

A perceptron is an artificial neuron that does certain computations to detect all the features in the input data. A perceptron combines with other perceptrons to compute any complex problems. This combination is called a Multi-layer Perceptron(MP) or Artificial Neural Network(ANN). MP necessarily consists of two layers namely input and output layers and a variable number of hidden layers. Based on the number of hidden layers MP can be classified into Deep or Shallow Neural networks. In general, a perceptron is made up of four parts: inputs, weights, sum/weighted sum, and activation function, which are briefly explained below:

- **Inputs:** These are values given to a perceptron for computing.

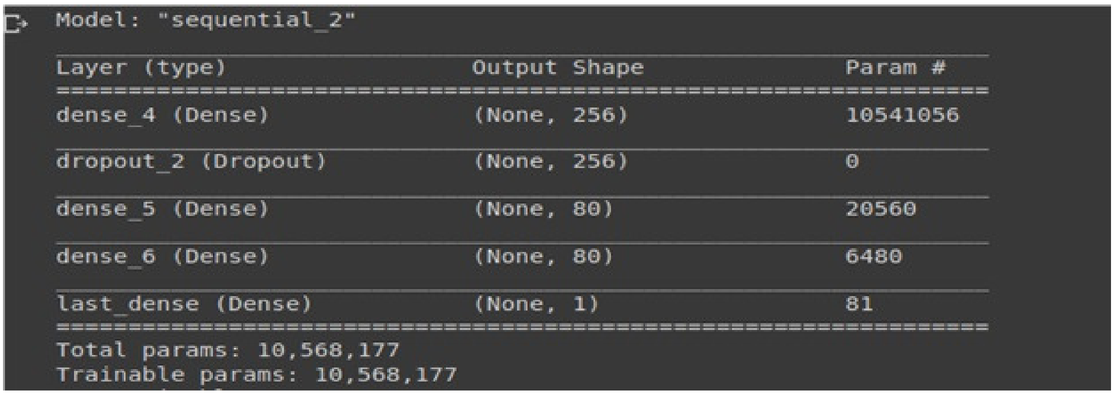
- **Weights:** Weights are the randomly initiated values stored in every neuron. These values mean the importance given to inputs for the computation.

- **Sum/Weighted Sum:** This is the sum of the product of input values and weights of that neuron. This is used to get an overall representation of the input values and is fed to the activation function.

- **Activation function:** Activation function is the important part of a neuron and gives the classification decision of that neuron. There are many activation functions and we used sigmoid and relu activation functions.

The basic concept of MP is its learning process using learning rate hyper parameters, back propagation, and bias. To begin with, we randomly initialize the values of all perceptrons. Now weighted sum is computed and bias is added to that and fed to the activation function. The output of the activation function which is also called a feed is forwarded to other nodes in the next layer. This is carried out till the feed reaches the output layer. The output from the output layer is checked with the actual value and the error is computed. If the error is less, then we conclude that the model is trained enough. Otherwise, the error is propagated back to the previous layer and is continued till it reaches the first layer. The weights are updated with a learning rate say alpha such that, the relative importance of the input values are adjusted. Now the weighted sum is computed and the feed is forwarded. The same process is carried out till the marginal error value is achieved.

Mathematically the process carried out at each layer is illustrated as follows. Let I be the input, W_1 is the randomly initiated weight matrix, H_1 is the output of the first hidden layer. Similarly, let W_2 is the randomly initiated weight matrix, H_2 is the output of the second hidden layer and W_3 is the randomly initiated weight matrix, H_3 is the output of the third hidden layer. Let O be the output of the output layer and W_4 be the weight of the output layer. Then the functioning at each layer is represented using the following equations.



```
Model: "sequential_2"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_4 (Dense)              (None, 256)               10541056
_________________________________________________________________
dropout_2 (Dropout)          (None, 256)               0
_________________________________________________________________
dense_5 (Dense)              (None, 80)                20560
_________________________________________________________________
dense_6 (Dense)              (None, 80)                6480
_________________________________________________________________
last_dense (Dense)           (None, 1)                 81
=================================================================
Total params: 10,568,177
Trainable params: 10,568,177
```

Figure 6.3: Multi-layer perceptron model

$$Firstlayer : H_1 = relu(I \times W_1) \tag{6.3}$$

$$Secondlayer : H_2 = relu(H_1 \times W_2) \tag{6.4}$$

$$Thirdlayer : H_3 = sigmoid(H_2 \times W_3) \tag{6.5}$$

$$Output : O = sigmoid(H_3 \times W_4) \tag{6.6}$$

Figure 6.3 depicts the model of multi-layer perceptron implemented along with the hyper parameters used. Here we developed a neural network with three hidden layers. Relu activation function is used for input and the first hidden layer. A sigmoid activation function is used for the second hidden layer and output layer.

6.4 EXPERIMENTAL DESCRIPTION

This section depicts the dataset along with evaluation metrics, implementation details, and experimental results. For multi-layer perceptron, we deduced the results with Word2Vec and GloVe word embeddings along with TF-IDF and count vectorizer. All models were prepared to utilize the hyper parameters with a procedure of 80 percent of the data for training and 20 percent for the test.

6.4.1 Dataset and Evaluation metrics

In (Ahmad et al. (2020)) authors trained a combination of different machine learning algorithms over four real-world datasets for fake news classification. From their experimental study, it was proved that the ISOT dataset has achieved a maximum accuracy

Table 6.1: Confusion matrix for Multinomial NB classifier with TF-IDF

	Predicted Positive	Predicted Negative
Actual Positive	1901	153
Actual Negative	227	1719

of 99% over other datasets. Hence this work adopted the ISOT data set for experimentation. The dataset consists of two CSV files. The first file named "True.csv" contains more than 12,600 articles from reuter.com. The second file named "Fake.csv" contains more than 12,600 articles from different fake news outlet resources. Each article contains information like article title, text, type, and the date the article was published on.

To evaluate the models, we used True positive rate, True negative rate and accuracy as the evaluation metrics and are described as follows:

- **True Positive Rate (TPR):** Percentage of a number of fake triples predicted correctly out of a total number of fake triples and mathematically defined as:
$$TPR = (\frac{True\ Positive(TP)}{TP + False\ Positive(FP)}) \times 100 \tag{6.7}$$

- **True Negative Rate (TNR):** Percentage of number of correctly predicted true news out of total number of true news and shown mathematically as:
$$TNR = (\frac{True\ Negative(TN)}{TN + False\ Negative(FN)}) \times 100 \tag{6.8}$$

- **Accuracy:** Percentage of fake and true news predicted correctly out of total number of news and mathematically shown as:
$$Accuracy = (\frac{TP + TN}{Total\ Number\ of\ News}) \times 100 \tag{6.9}$$

6.4.2 Implementation environment

The models are implemented using Google colab which is a jupyter notebook that runs in a cloud environment. Keras Python package is used to implement the machine learning-based baseline models and deep learning-based multi-layer perceptron model.

6.4.3 Evaluation of model performance

The TPR and TNR for the various machine learning algorithms can be analyzed by using a confusion matrix. For instance, the confusion matrix for multinomial NB using TF-IDF is as shown in table 6.1. In table 6.1, 1719 gives the number of times the model predicted the news as fake and news turned out to be fake, 1901 gives the number of times the model predicted the news as true and news turned out to be true, 153 gives the number of times the model predicted the news as true but the news turned out to be fake and 227 gives the number of times the model predicted the news as fake but the news turned out to be true.

Here, the models are trained with and without named entity tags, along with triples, and are evaluated separately to assess the effectiveness of using named entity tags. The Multi-layer Perceptron is tested by training the model with a combination of frequency and prediction-based word embeddings. The tables 6.2 and 6.3 show a consolidated view of performance metrics compared for implemented classifiers without and with named entity tags, respectively. From the tables, it is clear that the accuracy of models is slightly improved by considering named entity tags along with triples, as the named entity tags represent an additional feature for classification. The tables show that the MNB classifier has higher accuracy when using features extracted from the count vectorizer, as the NB classifiers do not work well with real numbers generated by TF-IDF models. Similarly, we can also infer that, of all the machine learning-based classifiers considered, the RF classifier provides the highest accuracy for features extracted using the TF-IDF model. As a result, RF is used as the baseline for comparison. RF, like Multi-layer Perceptron, is trained over vectorized triples using both frequency and prediction-based embedding models. The results show that Multi-layer Perceptron with count vectorizer and GloVe word embedding outperforms the baseline classifier in terms of accuracy. GloVe embedding produces higher accuracy as they capture more context-based semantics among triples than Word2Vec.

Based on the above experimental results, the following two hypothesis are defined and are verified using paired t-test, a statistical significant test to compare the performance of two classifiers on 10 test sets.

Table 6.2: Comparison of classifiers without named entity tags

Embedding	Classifier	Accuracy
Count Vectorizer	MNB	90.60
	PA	93.50
	SVM	84.02
	KNN	86.67
	LR	88.78
	RF	93.92
TF-IDF	MNB	85.00
	PA	94.02
	SVM	85.05
	KNN	87.56
	LR	89.34
	RF	94.25
Wrod2Vec	RF	95.02
GloVe		89.10
TF-IDF+Word2Vec		96.02
TF-IDF+GloVe		89.69
Count vectorizer+Word2Vec		90.93
Count vectorizer+GloVe		87.25
Wrod2Vec	Multi-layer Perceptron	95.98
GloVe		96.10
TF-IDF+Word2Vec		96.82
TF-IDF+GloVe		96.69
Count vectorizer+Word2Vec		97.03
Count vectorizer+GloVe		97.25

Table 6.3: Comparison of classifiers with named entity tags

Embedding	Classifier	Accuracy
Count Vectorizer	MNB	91.32
	PA	93.80
	SVM	84.85
	KNN	87.03
	LR	89.93
	RF	94.00
TF-IDF	MNB	85.66
	PA	94.35
	SVM	85.92
	KNN	88.34
	LR	90.12
	RF	94.95
Wrod2Vec	RF	95.96
GloVe		90.02
TF-IDF+Word2Vec		96.89
TF-IDF+GloVe		90.21
Count vectorizer+Word2Vec		91.34
Count vectorizer+GloVe		89.30
Wrod2Vec	Multi-layer Perceptron	96.76
GloVe		96.83
TF-IDF+Word2Vec		97.02
TF-IDF+GloVe		97.75
Count vectorizer+Word2Vec		97.92
Count vectorizer+GloVe		98.30

Table 6.4: Accuracy values for 10 test samples for the base line model and MLP model

	test set 1	**test set 2**	**test set 3**	**test set 4**	**test set 5**	**test set 6**	**test set 7**	**test set 8**	**test set 9**	**test set 10**
MLP	0.92	0.93	0.91	0.95	0.96	0.92	0.88	0.87	0.96	0.99
RF	0.91	0.92	0.92	0.93	0.89	0.93	0.86	0.90	0.97	0.98
diff	1	1	-1	2	7	-1	2	-3	-1	1

- H_0 - Null Hypothesis: A Multi-layer Perceptron classifier model built on count vectorizer and GloVe word embedding outperform an RF classifier built on TF-IDF and Word2Vec in predicting fake news.

- H_1 - Alternative Hypothesis: A Multi-layer Perceptron classifier model built on count vectorizer and GloVe word embedding does not outperform an RF classifier built on TF-IDF and Word2Vec in predicting fake news.

Here the RF classifier with TF-IDF and Word2Vec models and Multi-layer Perceptron with count vectorizer and GloVe models are trained and tested using a 10-fold cross-validation method. In this method, the accuracy scores of both the classifiers over ten folds are averaged. Each chunk of data is entered exactly once into the validation set and 9 times into the training set. This reduces underfitting because all of the data is used for fitting, and it reduces overfitting because all of the data is used in the validation set. The table 6.4 shows the accuracy values for the two classifiers for ten test sets. Based on the readings it is found that the t-statistic t value for 10 test sets is smaller than the $t_{(p/2),9}$ for 9 degrees of freedom for $pvalue = 0.05$, so we reject the Alternate Hypothesis and accept the Null Hypothesis.

Similarly, the accuracy of Multi-layer Perceptron with count vectorizer and GloVe word embedding is compared with two recent gold standard models Khan et al. (2021b) and Kaliyar et al. (2021). It is found that Multi-layer Perceptron with count vectorizer and GloVe word embedding produces accuracy which is near to the gold standard as the GloVe embedding captures more context-based semantics among triples.

6.5 SUMMARY

This chapter proposes a method for determining the trustworthiness of knowledge base facts using a deep learning-based Multi-layer Perceptron model. To the best of our knowledge, this experiment is the first attempt at developing a model to classify a given knowledge base fact or triple as false or true. For feature embedding, a data modeling approach is proposed using two state-of-the-art word embedding models, Word2Vec and GloVe, as well as traditional TF-IDF and Count vectorizer. Using both prediction and frequency-based word embedding models allows for the consideration of both word and document level features, as well as the reduction of vector dimension. Empirical results show that a Multi-layer Perceptron trained on triples vectorized with a count vectorizer and GloVe word embedding produced the highest accuracy of 98.3% over the baseline model. The baseline model is obtained by training six machine learning classifiers over the triples and selecting the one that gives the highest accuracy. It is found from the experiment that, the Random Forest classifier trained over triples, modeled with TF-IDF, produced the highest accuracy of 94.95 compared to the remaining machine learning classifiers. The empirical results also show that the accuracy of models is slightly improved by considering named entity tags along with triples, as the named entity tags represent an additional feature for classification. A statistical significance test is also conducted using paired t-test with 10-fold cross-validation to validate the performance evaluation test conducted.

CHAPTER 7

CONCLUSIONS AND FUTURE SCOPE

In this work, a crime knowledge base is generated by extracting and integrating the information from English and Hindi news articles. This is achieved in three phases. In the first phase, an initial version of KB is constructed from English news articles. Because of the common source for extraction such as DBpedia, the generation of erroneously tagged entities and hence the redundancy is avoided by the proposed algorithm. Similarly, when compared to un-tagged entities produced by NLTK, the proposed approach outperforms in tagging the entities with proper tags. The empirical results show that the proposed approach for information extraction achieves the highest accuracy of 96% for sentences with a single relation compared to the entities extracted from NLTK. However, the generation of un-tagged entities and hence the loss of information still exist if an entity can not link to any DBpedia property. This can be improved by introducing more background KBs in the future. This work also explores the use of both contextual as-well-as semantic similarity measures for finding the similarity between the entities. Empirical results show that redundancy is minimized to a greater extent by using both similarity measures.

In the second phase, a clustering and bootstrapping-based generic framework for knowledge base completion are proposed. Using the framework, any knowledge base created with the facts extracted from English news articles can be enriched with new facts available in low-resourced language articles without using language-specific tools. Here, the experiment is conducted using the low-resourced Indian language Hindi news articles. The redundancies that exist among the bi-lingual collection of articles are exploited by grouping the articles that are topically or sententially similar using the nearest neighborhood clustering. Empirical results show that clusters of high quality are obtained by representing the clusters using triples against the traditional Bag of Words representation. Empirical results show that the proposed algorithm takes more time compared to the baseline approach due to the extraction of two or more triples from a single headline. However, the clusters of high quality with an average value of 0.63 as silhouette coefficients are obtained for proposed algorithms using bi-lingual facts. The proposed algorithm is also evaluated for mono-lingual facts by comparing the results with two recently proposed works over Reuters and 20Newsgroup datasets. The proposed algorithm achieved the highest accuracy of 0.5210 and 0.4832 for Reuters and 20Newsgroup datasets respectively. However, due to the generation of more individual clusters i.e. clusters with a single element, the purity of the proposed algorithm is less compared to an existing work with N-grams representation along with K-means clustering with improved square root similarity measure. From each group of related articles, the facts related to English news articles are bootstrapped to extract the facts from Hindi news articles using Google translator API. This way of using the high-resource language facts as bootstrapping triples helps to extract the facts from articles related to the languages for which language processing tools like POS tags are neither available nor accurate. Experimental results for extraction show that using the framework a better recall of 93% is achieved in identifying the new facts compared to precision. This is evident from the fact that the total number of new facts extracted by the proposed approach is more due to improper projection of bootstrapping triples with the triples extracted from candidate sentences. We also ran 1000 triples through Googletrans, a python library for Google Translator to check the accuracy of google translator. The results are manually verified, and the translator's accuracy is 98.3 percent, with the exception of a few proper nouns, such as "Gurgaon", where the translation did not match.

For the third phase, a deep learning-based Multi-layer Perceptron model to identify the trustworthiness of knowledge base facts is implemented. To the best of our knowledge, this experiment is the first attempt at developing a model to classify a given knowledge base fact or triple as false or true. For feature embedding, a data modeling approach is proposed using two state-of-the-art word embedding models, Word2Vec and GloVe, as well as traditional TF-IDF and Count vectorizer. Using both prediction and frequency-based word embedding models allows for the consideration of both word and document level features, as well as the reduction of vector dimension. Empirical results show that a Multi-layer Perceptron trained on triples vectorized with a count vectorizer and GloVe word embedding produced the highest accuracy of 98.3% over the baseline model. The baseline model is obtained by training six machine learning classifiers over the triples and selecting the one that gives the highest accuracy. It is found from the experiment that, the Random Forest classifier trained over triples, modeled with TF-IDF, produced the highest accuracy of 94.95 compared to the remaining machine learning classifiers. The empirical results also show that the accuracy of models is slightly improved by considering named entity tags along with triples, as the named entity tags represent an additional feature for classification. A statistical significance test is also conducted using paired t-test with 10-fold cross-validation to validate the performance evaluation test conducted.

FUTURE SCOPE

1. **Triple extraction from sentences with multiple relations:** The present work considers the uni-relational sentences i.e. sentences with a single relation for extraction. Even though the experiment is conducted for sentences with two and three relations, a significant improvement in performance is achieved for uni-relational sentences. Moreover, the work assumes that a relationship must have non-empty subject and object entities. For instance, if a sentence starts with a relational term like "Walking on the rope, he said", the proposed solution generates invalid triples. In the future, it is most essential to improve the proposed solution to extract the triples from multi-relational sentences, irrespective of where the relationship is placed in the sentence.

2. **Anaphora Resolution:** The current system does not address the problem of replacing the pronoun called anaphora like he or she with an appropriate noun in

a sentence, especially when a sentence includes more than one noun. Moreover, anaphora resolution across the sentences and paragraphs is challenging which is to be considered in the future.

3. **Extraction from multi-modal and multi-lingual sources:** The present system considered only news articles with text and image modality data for extraction. Now a day's sources like social media, Live news-TV shows, Youtube, Instagram, and so on are not only a collection of textual data but also a pool of multi-media data like audio and video along with images with or without proper tags/annotations. This needs special attention to extract the multi-media data based on the tags and multi-lingual comments used by users or their friends.

4. **Validation of multi-modal and multi-lingual data:** The utilization of multi-modal data like images, audio, and video is also understudied in the field of fake news detection. The training of deep learning models jointly extracting the features from triples and the multi-modal data provides a big scope for knowledge-based systems that enrich the multi-modal facts into the knowledge base. Similarly, the utilization of multi-lingual data for fake news detection is also a challenge. From a knowledge base construction perspective, studying the behavior of a machine and deep learning models trained using triples extracted from multi-lingual sources is of utmost necessity in the global era of the internet where the knowledge extraction takes place from multi-lingual sources. This provides a new challenge to develop language-independent machines and deep learning models to classify knowledge base facts as true or fake. To summarize, even though a lot of works are published in the domain of fake news detection, binary classification of knowledge base facts into fake or true is in its infant stage.

5. **Compatibility with other KBs**: "Web of Linked Data" is the main vision of SW to link the extracted knowledge with well-established Linked Open Data (LOD) repositories like DBpedia. This contributes to global data integration and provides unlimited search and querying capabilities by sharing and accessing data across the web without the format and language restrictions. The format and

language-specific human readable representation of data can not be used by the machine directly to answer natural language queries like “Get me the name of the culprit involved in a particular crime”. A deep understanding of natural language texts is required to answer such queries. Hence it is desirable for a KB to store the knowledge extracted out of unstructured text in a machine-readable form like standard RDF which can be shared across the applications and community boundaries. KB enriched with RDF facts should be accessible through an appropriate query language like SPARQL. RDF facts generated for the current knowledge base are not globally sharable or accessible, which will be addressed in the future.

BIBLIOGRAPHY

Agarwal, A., Mittal, M., Pathak, A., and Goyal, L. M. (2020). Fake News Detection Using a Blend of Neural Networks: An Application of Deep Learning. *SN Computer Science*, 1(3):1–9.

Ahmad, I., Yousaf, M., Yousaf, S., and Ahmad, M. O. (2020). Fake news detection using machine learning ensemble methods. *Complexity*, 2020.

Akbik, A., Bergmann, T., Blythe, D., Rasul, K., Schweter, S., and Vollgraf, R. (2019). FLAIR: An easy-to-use framework for state-of-the-art NLP. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59, Minneapolis, Minnesota. Association for Computational Linguistics.

Akbik, A., Danilevsky, M., Kbrom, Y., Li, Y., Zhu, H., et al. (2016). Multilingual information extraction with polyglotie. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations*, pages 268–272.

Alani, H., Kim, S., Millard, D. E., Weal, M. J., Lewis, P. H., Hall, W., and Shadbolt, N. (2003). Automatic extraction of knowledge from web documents.

Albahr, A. and Albahar, M. (2020). An empirical comparison of fake news detection using different machine learning algorithms. *International Journal of Advanced Computer Science and Applications*, 11(9):146–152.

Aliprandi, C., Arraiza Irujo, J., Cuadros, M., Maier, S., Melero, F., and Raffaelli, M. (2014). Caper: Collaborative information, acquisition, processing, exploitation and reporting for the prevention of organised crime. In Stephanidis, C., editor, *HCI International 2014 - Posters' Extended Abstracts*, pages 147–152, Cham. Springer International Publishing.

Alruily, M., Ayesh, A., and Zedan, H. (2014). Crime profiling for the arabic language using computational linguistic techniques. *Information Processing & Management*, 50(2):315 – 341.

Arulanandam, R., Savarimuthu, B. T. R., and Purvis, M. A. (2014). Extracting crime information from online newspaper articles. In *Proceedings of the Second Australasian Web Conference - Volume 155*, AWC '14, pages 31–38, Darlinghurst, Australia, Australia. Australian Computer Society, Inc.

Augenstein, I., Maynard, D., and Ciravegna, F. (2016). Distantly supervised web relation extraction for knowledge base population. *Semantic Web*, 7(4):335–349.

Augenstein, I., Padó, S., and Rudolph, S. (2012). Lodifier: Generating linked data from unstructured text. In Simperl, E., Cimiano, P., Polleres, A., Corcho, O., and Presutti, V., editors, *The Semantic Web: Research and Applications*, pages 210–224, Berlin, Heidelberg. Springer Berlin Heidelberg.

Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998). The berkeley framenet project. In *Proceedings of the 17th international conference on Computational linguistics-Volume 1*, pages 86–90. Association for Computational Linguistics.

Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., and Etzioni, O. (2007). Open information extraction from the web. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, pages 2670–2676, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Bisandu, D. B., Prasad, R., and Liman, M. M. (2018). Clustering news articles using efficient similarity measure and n-grams. *International Journal of Knowledge Engineering and Data Mining*, 5(4):333–348.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Buitelaar, P., Cimiano, P., Frank, A., Hartung, M., and Racioppa, S. (2008). Ontology-based information extraction and integration from heterogeneous data sources. *International Journal of Human-Computer Studies*, 66(11):759 – 788.

Calijorne Soares, M. A. and Parreiras, F. S. (2020). A literature review on question answering techniques, paradigms and systems. *Journal of King Saud University - Computer and Information Sciences*, 32(6):635–646.

Candan, K. S., Liu, H., and Suvarna, R. (2001). Resource description framework: metadata and its applications. *ACM SIGKDD Explorations Newsletter*, 3(1):6–19.

Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka, Jr., E. R., and Mitchell, T. M. (2010). Toward an architecture for never-ending language learning. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*, AAAI'10, pages 1306–1313. AAAI Press.

Chau, M., Xu, J. J., and Chen, H. (2002). Extracting meaningful entities from police narrative reports. In *Proceedings of the 2002 Annual National Conference on Digital Government Research*, dg.o '02, pages 1–5. Digital Government Society of North America.

Chaudhuri, S., Ganti, V., and Kaushik, R. (2006). A primitive operator for similarity joins in data cleaning. In *ICDE*. Institute of Electrical and Electronics Engineers, Inc.

Chen, M., Tian, Y., Yang, M., and Zaniolo, C. (2017). Multilingual knowledge graph embeddings for cross-lingual knowledge alignment. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, IJCAI'17, page 1511–1517. AAAI Press.

Church, K. W. (2017). Word2vec. *Natural Language Engineering*, 23(1):155–162.

Cunningham, H., Humphreys, K., Gaizauskas, R., and Wilks, Y. (1997). Gate - a general architecture for text engineering. pages 29–30.

Dasgupta, T., Naskar, A., Saha, R., and Dey, L. (2017). Crimeprofiler: crime information extraction and visualization from news media. In *WI*.

Derczynski, L., Maynard, D., Rizzo, G., van Erp, M., Gorrell, G., Troncy, R., Petrak, J., and Bontcheva, K. (2015). Analysis of named entity recognition and linking for tweets. *Information Processing & Management*, 51(2):32 – 49.

Dhuria, S., Taneja, H., and Taneja, K. (2016). Nlp and ontology based clustering — an integrated approach for optimal information extraction from social web. In *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, pages 1765–1770.

Dong, X., Gabrilovich, E., Heitz, G., Horn, W., Lao, N., Murphy, K., Strohmann, T., Sun, S., and Zhang, W. (2014). Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 601–610, New York, NY, USA. ACM.

Dragos, V. (2013). Developing a core ontology to improve military intelligence analysis. *International Journal of Knowledge-based and Intelligent Engineering Systems*, 17(1):29–36.

Elavarasi, S. A., Akilandeswari, J., and Menaga, K. (2014). A survey on semantic similarity measure. *International Journal of Research in Advent Technology*, 2(3):389–398.

Erxleben, F., Günther, M., Krötzsch, M., Mendez, J., and Vrandečić, D. (2014). Introducing wikidata to the linked data web. In *International Semantic Web Conference*, pages 50–65. Springer.

Etzioni, O., Cafarella, M., Downey, D., Kok, S., Popescu, A.-M., Shaked, T., Soderland, S., Weld, D. S., and Yates, A. (2004). Web-scale information extraction in knowitall: (preliminary results). In *Proceedings of the 13th International Conference on World Wide Web*, WWW '04, pages 100–110, New York, NY, USA. ACM.

Fader, A., Soderland, S., and Etzioni, O. (2011). Identifying relations for open information extraction. In *Proceedings of the Conference on Empirical Methods in Natural*

Language Processing, EMNLP ’11, pages 1535–1545, Stroudsburg, PA, USA. Association for Computational Linguistics.

Fadhil, A. and Haarslev, V. (2006). Gloo: A graphical query language for owl ontologies. In *OWLED*, volume 216.

Fadhil, A. and Haarslev, V. (2007). Ontovql: A graphical query language for owl ontologies. In *Description Logics*.

Fan, J., Kalyanpur, A., Gondek, D. C., and Ferrucci, D. A. (2012). Automatic knowledge extraction from documents. *IBM Journal of Research and Development*, 56(3.4):5:1–5:10.

Furtado, V., Ayres, L., de Oliveira, M., Vasconcelos, E., Caminha, C., D’Orleans, J., and Belchior, M. (2010). Collective intelligence in law enforcement – the wikicrimes system. *Information Sciences*, 180(1):4 – 17. Special Issue on Collective Intelligence.

Gamallo, P. and Garcia, M. (2015). Multilingual open information extraction. In *Portuguese Conference on Artificial Intelligence*, pages 711–722. Springer.

Gangemi, A. (2013). A comparison of knowledge extraction tools for the semantic web. In *Extended semantic web conference*, pages 351–366. Springer.

Gangemi, A., Presutti, V., Recupero, D. R., Nuzzolese, A. G., Draicchio, F., and Mongiovì, M. (2017). Semantic web machine reading with fred. *Semantic Web*, 8:873–893.

Gerber, D., Hellmann, S., Bühmann, L., Soru, T., Usbeck, R., and Ngonga Ngomo, A.-C. (2013). Real-time rdf extraction from unstructured data streams. In Alani, H., Kagal, L., Fokoue, A., Groth, P., Biemann, C., Parreira, J. X., Aroyo, L., Noy, N., Welty, C., and Janowicz, K., editors, *The Semantic Web – ISWC 2013*, pages 135–150, Berlin, Heidelberg. Springer Berlin Heidelberg.

Gerber, D. and Ngomo, A.-C. N. (2012). Extracting multilingual natural-language patterns for rdf predicates. In ten Teije, A., Völker, J., Handschuh, S., Stuckenschmidt, H., d’Acquin, M., Nikolov, A., Aussenac-Gilles, N., and Hernandez, N., editors, *Knowledge Engineering and Knowledge Management*, pages 87–96, Berlin, Heidelberg. Springer Berlin Heidelberg.

Ghosh, S. and Shah, C. (2018). Towards automatic fake news classification. *Proceedings of the Association for Information Science and Technology*, 55(1):805–807.

Ghosh, S. and Shah, C. (2019). Toward Automatic Fake News Classification. *Proceedings of the 52nd Hawaii International Conference on System Sciences*, 6:2254–2263.

Goyal, A., Gupta, V., and Kumar, M. (2018). Recent named entity recognition and classification techniques: A systematic review. *Computer Science Review*, 29:21 – 43.

Gupta, S. and Meel, P. (2021). Fake news detection using passive-aggressive classifier. In Ranganathan, G., Chen, J., and Rocha, Á., editors, *Inventive Communication and Computational Technologies*, pages 155–164, Singapore. Springer Singapore.

Haase, P., Broekstra, J., Eberhart, A., and Volz, R. (2004). A comparison of rdf query languages. In *International Semantic Web Conference*, pages 502–517. Springer.

Hazrina, S., Sharef, N. M., Ibrahim, H., Murad, M. A. A., and Noah, S. A. M. (2017). Review on the advancements of disambiguation in semantic question answering system. *Information Processing & Management*, 53(1):52 – 69.

Hecking, M. and Schwerdt, C. (2008). Multilingual information extraction for intelligence purposes. In *Proceedings of the 13th International Command and Control Research and Technolgy Symposium (ICCRTS).*

Heindorf, S., Potthast, M., Stein, B., and Engels, G. (2016). Vandalism detection in wikidata. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 327–336.

Heist, N., Hertling, S., Ringler, D., and Paulheim, H. (2020). Knowledge graphs on the web – an overview.

Hoffart, J., Suchanek, F. M., Berberich, K., and Weikum, G. (2013). Yago2: A spatially and temporally enhanced knowledge base from wikipedia. *Artificial Intelligence*, 194:28–61.

Hossein, H., Xu, H., S., S. E., and Mansi, G. A review of data mining applications in crime. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 9(3):139–154.

Islam, A. and Inkpen, D. (2008). Semantic text similarity using corpus-based word similarity and string similarity. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 2(2):1–25.

Jalil, M. . M. A., Ling, C. P., Noor, N. M. M., and Mohd., F. (2017). Knowledge representation model for crime analysis. *Procedia Computer Science*, 116:484 – 491. Discovery and innovation of computer science technology in artificial intelligence era: The 2nd International Conference on Computer Science and Computational Intelligence (ICCSCI 2017).

Ji, S., Pan, S., Cambria, E., Marttinen, P., and Philip, S. Y. (2021). A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE Transactions on Neural Networks and Learning Systems*.

Jia, S., Xiang, Y., Chen, X., and Wang, K. (2019). Triple trustworthiness measurement for knowledge graph. *The World Wide Web Conference on - WWW '19*.

Kaliyar, R. K. (2018). Fake news detection using a deep neural network. In *2018 4th International Conference on Computing Communication and Automation (ICCCA)*, pages 1–7.

Kaliyar, R. K., Goswami, A., and Narang, P. (2021). FakeBERT: Fake news detection in social media with a BERT-based deep learning approach. *Multimedia Tools and Applications*, 80(8):11765–11788.

Khan, J. Y., Khondaker, M. T. I., Afroz, S., Uddin, G., and Iqbal, A. (2021a). A benchmark study of machine learning models for online fake news detection. *Machine Learning with Applications*, 4:100032.

Khan, J. Y., Khondaker, M. T. I., Afroz, S., Uddin, G., and Iqbal, A. (2021b). A benchmark study of machine learning models for online fake news detection. *Machine Learning with Applications*, 4:100032.

Klein, P., Ponzetto, S. P., and Glavaš, G. (2017). Improving neural knowledge base completion with cross-lingual projections. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 516–522, Valencia, Spain. Association for Computational Linguistics.

Kondrak, G. (2005). N-gram similarity and distance. In *Proceedings of the 12th International Conference on String Processing and Information Retrieval*, SPIRE'05, pages 115–126, Berlin, Heidelberg. Springer-Verlag.

Ku, C. H., Iriberri, A., and Leroy, G. (2008). Natural language processing and e-government: Crime information extraction from heterogeneous data sources. In *Proceedings of the 2008 International Conference on Digital Government Research*, dg.o '08, pages 162–170. Digital Government Society of North America.

Kubias, A., Schenk, S., Staab, S., and Pan, J. Z. (2007). Owl saiql - an owl dl query language for ontology extraction. In *OWLED*.

Kuck, G. (2004). Tim berners-lee's semantic web. *South African Journal of information management*, 6(1).

Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P. N., Hellmann, S., Morsey, M., Van Kleef, P., Auer, S., et al. (2015). Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195.

Liang, J., Xiao, Y., Zhang, Y., Hwang, S.-w., and Wang, H. (2017). Graph-based wrong isa relation detection in a large-scale lexical taxonomy. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Loper, E. and Bird, S. (2002). Nltk: The natural language toolkit. In *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural*

Language Processing and Computational Linguistics - Volume 1, ETMTNLP '02, pages 63–70, Stroudsburg, PA, USA. Association for Computational Linguistics.

Louhi, I., Boudjeloud-Assala, L., and Tamisier, T. (2016). Incremental nearest neighborhood graph for data stream clustering. pages 2468–2475.

Malaviya, C., Bhagavatula, C., Bosselut, A., and Choi, Y. (2020). Commonsense knowledge base completion with structural and semantic context. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 2925–2933.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Inc, P., Bethard, S. J., and Mcclosky, D. (2014). The stanford corenlp natural language processing toolkit. In *In ACL, System Demonstrations*.

Manzoor, S. I., Singla, J., et al. (2019). Fake news detection using machine learning approaches: A systematic review. In *2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*, pages 230–234. IEEE.

Martinez-Gil, J. (2015). Automated knowledge base management: A survey. *Computer Science Review*, 18:1–9.

Martinez-Rodriguez, J. L., Hogan, A., and Lopez-Arevalo, I. (2018). Information extraction meets the semantic web: A survey. *Semantic Web*, (Preprint):1–81.

McGuinness, D. L., Van Harmelen, F., et al. (2004). Owl web ontology language overview. *W3C recommendation*, 10(10):2004.

Mihalcea, R., Corley, C., Strapparava, C., et al. (2006). Corpus-based and knowledge-based measures of text semantic similarity. In *Aaai*, volume 6, pages 775–780.

Miller, D. R. H., Leek, T., and Schwartz, R. M. (1999). A hidden markov model information retrieval system. In *Proceedings of the 22Nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '99, pages 214–221, New York, NY, USA. ACM.

Miller, F. P., Vandome, A. F., and McBrewster, J. (2009). *Levenshtein Distance: Information Theory, Computer Science, String (Computer Science), String Metric, Damerau?Levenshtein Distance, Spell Checker, Hamming Distance*. Alpha Press.

Miller, G. A. (1995). Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Milne, D. N., Witten, I. H., and Nichols, D. M. (2007). A knowledge-based search engine powered by wikipedia. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 445–454.

Moschitti, A., Tymoshenko, K., Alexopoulos, P., Walker, A., Nicosia, M., Vetere, G., Faraotti, A., Monti, M., Pan, J. Z., Wu, H., and Zhao, Y. (2017). *Question Answering and Knowledge Graphs*, pages 181–212. Springer International Publishing, Cham.

Nakashole, N., Theobald, M., and Weikum, G. (2011). Scalable knowledge harvesting with high precision and high recall. In *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*, WSDM '11, pages 227–236, New York, NY, USA. ACM.

Nakashole, N., Weikum, G., and Suchanek, F. (2013). Discovering semantic relations from the web and organizing them with patty. *SIGMOD Rec.*, 42(2):29–34.

Navigli, R. and Ponzetto, S. P. (2012). Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

Nickel, M., Tresp, V., and Kriegel, H.-P. (2011). A three-way model for collective learning on multi-relational data. In *Icml*.

Oramas, S., Espinosa-Anke, L., Sordo, M., Saggion, H., and Serra, X. (2016). Information extraction for knowledge base construction in the music domain. *Data & Knowledge Engineering*, 106:70 – 83.

Otegi, A., Ansa, O., and Agirre, E. (2015). Using knowledge-based relatedness for information retrieval. *Knowledge and Information Systems*, 44:689–718.

Paulheim, H. (2017). Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic web*, 8(3):489–508.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Pezeshkpour, P., Tian, Y., and Singh, S. (2020). Revisiting evaluation of knowledge base completion models. In *Automated Knowledge Base Construction*.

Poli, R., Healy, M., and Kameas, A. *Theory and applications of ontology: Computer applications*. Springer.

Popov, B., Kiryakov, A., Ognyanoff, D., Manov, D., Kirilov, A., and Goranov, M. (2003). Towards semantic web information extraction. In *Human Language Technologies Workshop at the 2nd International Semantic Web Conference (ISWC2003)*, volume 20.

Priyanga, V., Sanjanasri, J., Menon, V. K., Gopalakrishnan, E., and Soman, K. (2021). Exploring fake news identification using word and sentence embeddings. *Journal of Intelligent & Fuzzy Systems*, (Preprint):1–8.

Qu, R., Fang, Y., Bai, W., and Jiang, Y. (2018). Computing semantic similarity based on novel models of semantic representation using wikipedia. *Information Processing & Management*, 54(6):1002 – 1021.

Rashidghalam, H., Taherkhani, M., and Mahmoudi, F. (2016). Text summarization using concept graph and babelnet knowledge base. In *2016 Artificial Intelligence and Robotics (IRANOPEN)*, pages 115–119.

Rebele, T., Suchanek, F., Hoffart, J., Biega, J., Kuzey, E., and Weikum, G. (2016). Yago: A multilingual knowledge base from wikipedia, wordnet, and geonames. In *International Semantic Web Conference*, pages 177–185. Springer.

Reddy, H., Raj, N., Gala, M., and Basava, A. (2020). Text-mining-based Fake News Detection Using Ensemble Methods. *International Journal of Automation and Computing*, 17(2):210–221.

Rospocher, M., van Erp, M., Vossen, P., Fokkens, A., Aldabe, I., Rigau, G., Soroa, A., Ploeger, T., and Bogaard, T. (2016). Building event-centric knowledge graphs from news. *Web Semantics: Science, Services and Agents on the World Wide Web*, 37-38:132 – 151.

Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53 – 65.

Rozenfeld, B. and Feldman, R. (2008). Self-supervised relation extraction from the web. *Knowledge and Information Systems*, 17(1):17–33.

Salton, G. and Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, 24(5):513 – 523.

Saruladha, K., Aghila, G., and Raj, S. (2010). A survey of semantic similarity methods for ontology based information retrieval. In *2010 Second International Conference on Machine Learning and Computing*, pages 297–301. IEEE.

Óscar Ferrández, Izquierdo, R., Ferrández, S., and Vicedo, J. L. (2009). Addressing ontology-based question answering with collections of user queries. *Information Processing & Management*, 45(2):175 – 188.

Schuler, K. K. (2005). Verbnet: A broad-coverage, comprehensive verb lexicon.

Shi, B. and Weninger, T. (2016). Discriminative predicate path mining for fact checking in knowledge graphs. *Knowledge-based systems*, 104:123–133.

Shu, K., Cui, L., Wang, S., Lee, D., and Liu, H. (2019). Defend: Explainable fake news detection. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 395–405.

Shu, K., Sliva, A., Wang, S., Tang, J., and Liu, H. (2017). Fake news detection on social media: A data mining perspective. *ACM SIGKDD explorations newsletter*, 19(1):22–36.

Sirin, E. and Parsia, B. (2007). Sparql-dl: Sparql query for owl-dl. In *In 3rd OWL Experiences and Directions Workshop (OWLED-2007*.

Sohangir, S. and Wang, D. (2017). Improved sqrt-cosine similarity measurement. *Journal of Big Data*, 4(1):1–13.

Stern, R. and Sagot, B. (2012). Population of a knowledge base for news metadata from unstructured text and web data. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction*, AKBC-WEKEX '12, pages 35–40, Stroudsburg, PA, USA. Association for Computational Linguistics.

Thota, A., Tilak, P., Ahluwalia, S., and Lohia, N. (2018). Fake News Detection: A Deep Learning Approach. *SMU Data Science Review*, 1(3):1–20.

Timofeyev, A. and Choi, B. (2018). Building a knowledge based summarization system for text data mining. pages 118–133.

Trentelman, K. (2009). Survey of knowledge representation and reasoning systems.

Tsuruoka, Y., McNaught, J., Tsujii, J., and Ananiadou, S. (2007). Learning string similarity measures for gene/protein name dictionary look-up using logistic regression. *Bioinformatics*, 23(20):2768–2774.

Wan, S. and Angryk, R. A. (2007). Measuring semantic similarity using wordnet-based context vectors. In *2007 IEEE International Conference on Systems, Man and Cybernetics*, pages 908–913.

Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612.

Wiedemann, G., Yimam, S. M., and Biemann, C. (2018). A multilingual information extraction pipeline for investigative journalism. *arXiv preprint arXiv:1809.00221*.

Wu, Z., Chen, L., and Giles, C. L. (2015). Storybase: Towards building a knowledge base for news events. In *ACL*.

Wu, Z. and Palmer, M. (1994). Verbs semantics and lexical selection. In *Proceedings of the 32Nd Annual Meeting on Association for Computational Linguistics*, ACL '94, pages 133–138, Stroudsburg, PA, USA. Association for Computational Linguistics.

Xiao, C., Wang, W., and Lin, X. (2008). Ed-join: An efficient algorithm for similarity joins with edit distance constraints. *Proc. VLDB Endow.*, 1(1):933–944.

Xie, S. and Liu, Y. (2008). Using corpus and knowledge-based similarity measure in maximum marginal relevance for meeting summarization. In *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4985–4988. IEEE.

Zelenko, D., Aone, C., and Richardella, A. (2003). Kernel methods for relation extraction. *Journal of machine learning research*, 3(Feb):1083–1106.

Zhang, J., Dong, B., and Philip, S. Y. (2020). Fakedetector: Effective fake news detection with deep diffusive neural network. In *2020 IEEE 36th International Conference on Data Engineering (ICDE)*, pages 1826–1829. IEEE.

Zhou, X., Zafarani, R., Shu, K., and Liu, H. (2019). Fake news: Fundamental theories, detection strategies and challenges. In *Proceedings of the twelfth ACM international conference on web search and data mining*, pages 836–837.

Zhu, G. and Iglesias, C. A. (2018). Exploiting semantic similarity for named entity disambiguation in knowledge graphs. *Expert Systems with Applications*, 101:8 – 24.

Zouaq, A., Gagnon, M., and Jean-Louis, L. (2017). An assessment of open relation extraction systems for the semantic web. *Information Systems*, 71:228 – 239.

PUBLICATIONS

1. Srinivasa K, P. Santhi Thilagam. Crime base: Towards building a knowledge base for crime entities and their relationships from online news papers. Information Processing & Management, Volume 56, Issue 6, 2019, 102059, ISSN 0306-4573. DOI: https://doi.org/10.1016/j.ipm.2019.102059

2. Srinivasa K, P. Santhi Thilagam. Clustering and Bootstrapping Based Framework for News Knowledge Base Completion. COMPUTING AND INFORMATICS, Volume 40, Issue 2, 2021, 318–340. DOI: https://doi.org/10.31577/cai_2021_2_318

3. Srinivasa K, P. Santhi Thilagam. Multi-layer Perceptron based Fake News Classification using Knowledge Base Triples. Applied Intelligence, Springer, 2021. (Under Review)

BIODATA

Name:	Srinivasa . K
Date of Birth:	18/04/1983
Gender:	Male
Marital Status:	Married
Father's Name:	K. Prahlada Rao
Mother's Name:	K. Shantha Bai
Permanent Address:	c/o Karur Nagaraj, #29, "Sri Padma Nilaya", Sai Ram Colony, Raghavendra Colony 2nd Stage, Bellary, Karnataka-582103
E-mail:	srinivas.karur@gmail.com
Mobile:	9845399083
Qualification:	B.E in Computer Science and Engineering (Vijayanagara Engineering College, Bellary) M.Tech in Computer Science and Engineering (National Institute of Technology-Karnataka, Surathkal)
Teaching Experience:	1) Lecturer from March 2005 to December 2010 2) Assistant Professor from January 2011 to Till date Department of Computer Science and Engineering, Siddaganga Institute of Technology, Tumakuru, Karnataka.
Areas of Interest:	Information Extraction, Natural Language Processing