Identifying Semantically Similar questions using NLP techniques and Linked Data Principles

Anirban Bhattacharjee

A Dissertation

Presented to the University of Dublin, Trinity College

in partial fulfilment of the requirements for the degree of

Master of Science in Computer Science (Intelligent Systems)

Supervisor: Professor Declan O'Sullivan

August 2019

Declaration

I, the undersigned, declare that this work has not previously been submitted as an exercise for a degree at this, or any other University, and that unless otherwise stated, is my own work.

Anirban Bhattacharjee

August 7, 2019

Permission to Lend and/or Copy

I, the undersigned, agree that Trinity College Library may lend or copy this thesis upon request.

Anirban Bhattacharjee

August 7, 2019

Acknowledgments

...ACKNOWLEDGE.....

ANIRBAN BHATTACHARJEE

University of Dublin, Trinity College
August 2019

Identifying Semantically Similar questions using NLP techniques and Linked Data Principles

Anirban Bhattacharjee, Master of Science in Computer Science

University of Dublin, Trinity College, 2019

Supervisor: Professor Declan O'Sullivan

In Community Question Answering (CQA) sites, despite active participation, a significant amount of questions on such sites remain unanswered due to a lot of reasons such as the question being poorly formed/worded, unavailability of any answerer or the increased inflow of questions in the same area which disinterests an answerer to answer the same question or having to redirect them multiple times to already answered questions. This research is an attempt to study if unstructured data is converted to structured data using state-of-the art natural language processing (NLP) techniques and Linked Data technologies, to what extent it could help a user in identifying semantically similar questions. One of the most contested themes in Computer Science is the ability to automatically map natural language semantics into programming languages. This research work is distinguished from other studies as we approach the problem from an ontology centred view and the idea of knowledge reuse forms the notion of this work. We evaluate our approach and open discussions on new ways to evaluate the identification of semantically similar questions . The key findings of this research demonstrate that using NLP techniques and Linked Data principles identification of semantically similar questions is viable. The proposed approach has a small but significant impact which can be leveraged for designing data models for the task of finding semantically similar questions.

Contents

List of Tables

List of Figures

Chapter 1

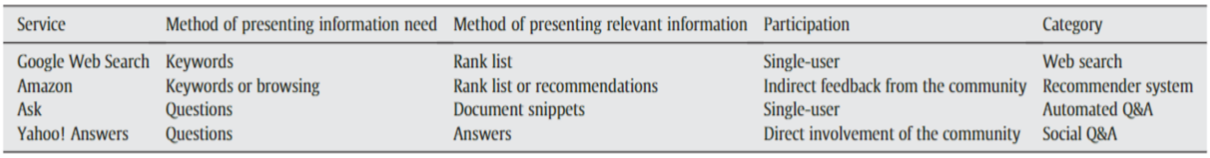
Introduction

Community Question Answering(CQA) sites is a platform for submitting questions in natural language, a forum for users to respond to questions in natural language and a community built around this exchange of questions and answers. CQA forums have emerged as an effective means of information and knowledge exchange on the Web. Over the years these CQA sites have accumulated a large number of user questions and answers and their related knowledge which forms the basis of rich knowledge repositories.CQA sites differ from the traditional web search in the following ways:

1. Instead of a list of documents from the web, users receive response in natural language to their questions.

2. Based on personal experience and expertise, answers to opinion or decision based questions can be sought in CQA sites, which cannot be retrieved through traditional web search.

The different online information seeking services and their categorization are listed in Figure 1.1 presented in the work of Shah et al. [2].



Service	Method of presenting information need	Method of presenting relevant information	Participation	Category
Google Web Search	Keywords	Rank list	Single-user	Web search
Amazon	Keywords or browsing	Rank list or recommendations	Indirect feedback from the community	Recommender system
Ask	Questions	Document snippets	Single-user	Automated Q&A
Yahoo! Answers	Questions	Answers	Direct involvement of the community	Social Q&A

Figure 1.1: Example of different information seeking services and their categorization [2]

Shah et al. [2] states that CQA sites requires a forum for public interaction, an interface to put forward an information need in the form of a natural language question as opposed to keyword searching and a community in which transaction of information is based on participation level.

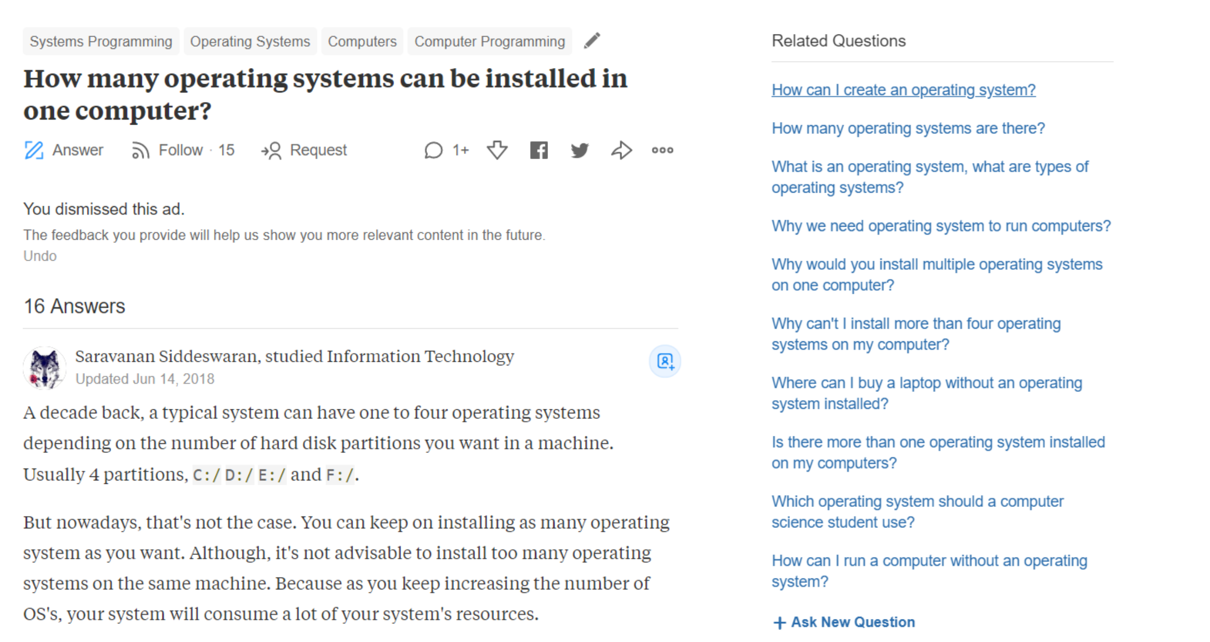
There are three broad categories of audience in these forums [9]:

1. Users who only ask questions.

2. Users who only answers questions.

3. Users who ask as well as answer questions.

The following Fig 1.2 illustrates a page from the CQA site.



How many operating systems can be installed in one computer?

You dismissed this ad.

The feedback you provide will help us show you more relevant content in the future.

Undo

16 Answers

Saravanan Siddeswaran, studied Information Technology

Updated Jun 14, 2018

A decade back, a typical system can have one to four operating systems depending on the number of hard disk partitions you want in a machine. Usually 4 partitions, `C:/` `D:/` `E:/` and `F:/`.

But nowadays, that's not the case. You can keep on installing as many operating system as you want. Although, it's not advisable to install too many operating systems on the same machine. Because as you keep increasing the number of OS's, your system will consume a lot of your system's resources.

How can I create an operating system?

How many operating systems are there?

What is an operating system, what are types of operating systems?

Why we need operating system to run computers?

Why would you install multiple operating systems on one computer?

Why can't I install more than four operating systems on my computer?

Where can I buy a laptop without an operating system installed?

Is there more than one operating system installed on my computers?

Which operating system should a computer science student use?

How can I run a computer without an operating system?

Figure 1.2: Example of a page from Quora CQA site [1]

CQA sites although sounds very exciting involves the risk of providing information to the users which are of significantly low quality[10].There are also instances when the content in these sites are abusive where both questions and answers are filled with either promotional content or offensive content.

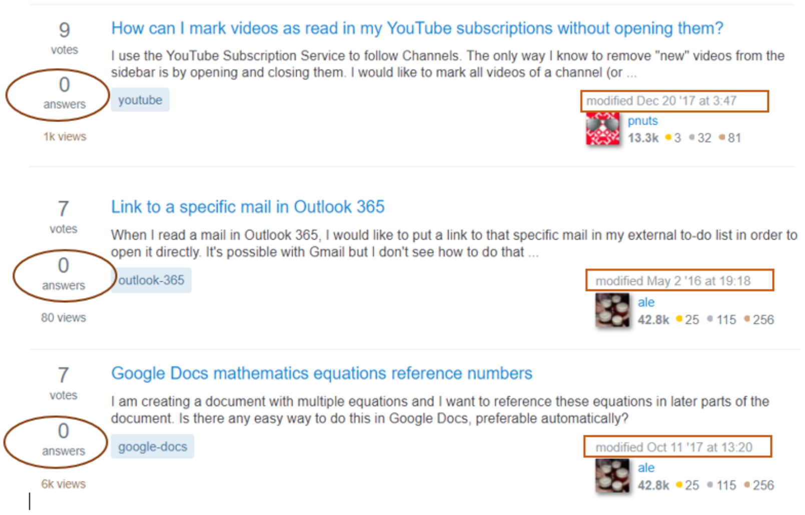
With the increasing popularity of CQA sites, it brings forward few challenges. According to relevant literature, most of the user generated content on CQA sites are redundant[11] and noisy [12][13].

1.1 Unanswered Questions in CQA sites

The usefulness of a CQA site depends on how effectively it can get answers for the questions posted by the users. In an anlaysis conducted by Shtok et al.[14] on Yahoo! answers which is one of the

[1] https://www.quora.com/How-many-operating-systems-can-be-installed-in-one-computer

very first CQA sites on the web, they found out that 15% of the questions remains unanswered which leaves the asker unsatisfied [14]. In order to observe the state of the questions after two days from the date of their posting in CQA sites, Li et al [15] randomly tracked 3000 recently posted questions in two CQA sites namely Yahoo! Answers and Baidu Zhidao. It was found that only 17.6% of questions in Yahoo! Answers received an answer within 48 hours. In Baidu Zhidao, only 22.7% questions are resolved [15]. These observations indicated that two of the major CQA sites couldn't solve user questions efficiently. Most of the CQA sites in this manner suffers from the problem of ”Answer Starvation”- a phenomenon where the users enjoy asking questions instead of answering them [16] which results in a large number of unanswered questions. According to [17] 12% of the questions in CQA sites are unanswered because of a duplicate question and 21.75% of questions are unanswered as it fails to attract an expert member. Research in this area [17] also signifies that the duration of an unanswered question differs from site to site. Figure 1.3 is an example of unanswered questions in CQA sites for over 24 months. The duration of a question remaining unanswered in Yahoo! is different than the duration of an unanswered question in Quora. Users registered in these sites may answer only a limited number of questions each day depending on their interests, determination and reputation. Although, there are active participants in CQA sites, a significant amount of questions remains unanswered due to a variety of reasons such as uninteresting questions, un-understandable questions, volume of incoming questions which creates a situation for answerers to overlook questions and the absence of a robust mechanism to redirect questions to potential answerers.



9 votes

0 answers

1k views

How can I mark videos as read in my YouTube subscriptions without opening them?

I use the YouTube Subscription Service to follow Channels. The only way I know to remove "new" videos from the sidebar is by opening and closing them. I would like to mark all videos of a channel (or ...

youtube

modified Dec 20 '17 at 3:47

pnuts

13.3k ●3 ●32 ●81

7 votes

0 answers

80 views

Link to a specific mail in Outlook 365

When I read a mail in Outlook 365, I would like to put a link to that specific mail in my external to-do list in order to open it directly. It's possible with Gmail but I don't see how to do that ...

outlook-365

modified May 2 '16 at 19:18

ale

42.8k ●25 ●115 ●256

7 votes

0 answers

6k views

Google Docs mathematics equations reference numbers

I am creating a document with multiple equations and I want to reference these equations in later parts of the document. Is there any easy way to do this in Google Docs, preferable automatically?

google-docs

modified Oct 11 '17 at 13:20

ale

42.8k ●25 ●115 ●256

Figure 1.3: Example of Unanswered Questions for over two years.

Estimating user expertise on a topic and re-directing the question to knowledge experts via inference engines as shown in Figure 1.4 could be done in a variety of ways. Research regarding this usually involves link analysis and topic modelling techniques. In relevant research literature, Jurczyk et al. [18] studied and proposed a graph modelling approach based on link analysis to calculate authoritative scores of users on the expected topics. Qu et al.[19] adopted the Probabilistic Latent Semantic Analysis (PLSA) model to capture user interactions and based on the user's history deduced correlation between answer and question, thereby improving the quality of recommending questions.

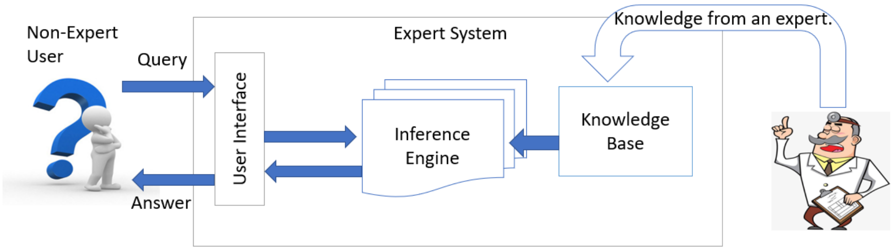


Figure 1.4: A flow chart of expert systems in CQA sites

For routing questions to most potential answerer a significant amount of research has been conducted [15] [20] [21] however the duration or the availability of the answerer to answer a recommended question is not defined. There are ways of generating answers from knowledge sources such as Wikipedia which has been discussed in relevant literature [21]. Although this approach being faster would need robust infrastructure for web scraping, data extraction and text summarization which are huge opportunities for research and are wide research fields. In this work, we research on a novel approach by using previously asked questions in CQA sites to identify semantically similar questions.

Dissatisfaction of the users could be combated by identifying semantically similar questions. It can also be used for recommending potential questions to users who are searching for an answer to a specific question and are not aware that such a question has already been answered in the past.

1.2 Motivation

Tim Berners-Lee, the prime inventor of the web and the semantic web initiator, the one who still leads the World Wide Web Consortium (W3C) is the first individual to publicly bring forward the need for semantics in the web at a World Wide Web (WWW) conference in the year 1994 [2]. He later proposed a road map for it's architecture at a very high level in the year 1998 [3]. The semantic web vision was primarily popularized when an article was published in the year 2001 [22]. This article specifically

[2] http://www.w3.org/Talks/WWW94Tim/
[3] https://www.w3.org/DesignIssues/Semantic.html

attracted a lot of researchers to realize this vision. Web was mostly a large graph of web pages back in the year 2001, however, when the semantic web vision was proposed, the need to represent semantic information aroused. Similar to the link-ability of documents on the web, the semantic web's most distinguished feature is the ability to reference specific pieces of data published by others. This web of data in contrast to the traditional web of documents can be accessed by both humans as well as machines. The standards and practices suggested by the W3C to publish data on the web is referred to as the *"Linked Data"*. Machines usually understand information which is provided in the form of linked data as data models are usually specified by ontologies which enables automated reasoning. Connecting distributed pieces of information related to disparate datasets are favoured by linked data.

Our intuition to use linked data to identify semantically similar questions stems from the opportunity to make data interconnected by using the collective wisdom of the queries of real world people in CQA sites. Rather than considering every word as a random string of characters, considering words as real things in the world and connecting those things to other things to aid an user in finding questions for which they are actually looking for an answer is the reason for attempting this approach.

Semantic Web is an extension of the World Wide Web in which the meaning(semantics) of information and services is defined, making it possible to understand and satisfy the requests of people and machines to use the web content. Linked data(LD) is at the heart of what Semantic Web is all about : large scale integration of, and reasoning on, data on the Web [4]. Semantic Web is about making links such that a person or a machine can explore the web of data. LD refers to a set of best practices for exposing, sharing, and connecting data on the Web. LD gives the power to re-use information in unexpected ways which can add value to a web of data. The information scale of unstructured data can make question answering systems over LD much more useful. Also, using structured resources to support information extraction for question answering or in this context finding similar questions will support the enrichment of structured datasets from unstructured data.

1.3 Research Question

To what extent is it possible to convert natural language questions into structured data (RDF graphs) and thereby querying RDF graphs using SPARQL protocol and RDF Query Language (SPARQL) to get semantically similar questions?

1.4 Contributions

In this dissertation - We propose a unique approach to identify semantically similar questions in CQA sites. In our approach we populate a knowledge base with questions posted in CQA sites. We create

[4] https://www.w3.org/standards/semanticweb/data

a Resource Description Framework (RDF) graph using the semantic triples generated from a natural language sentence. The nodes of the graph are either subjects or objects and the edges of the graph represent the relationship between two nodes. In this manner, unstructured natural language questions are modelled to structured sources of information in RDF data model .The triple store / RDF repository / knowledge base is then queried using SPARQL Protocol and RDF Query Language(SPARQL) to identify semantically similar questions based on the triples associated with each question. In the event of an unknown question posted by the users, the triples generated from the text of the question are used to populate the knowledge base and stored for future querying.

1.5 Organization of Dissertation

The rest of the dissertation is organized as follows: Chapter 2, discusses the relevant background including the state of the art for identifying semantically similar questions. Chapter 3, discusses our attempt to design a proof of concept application using state-of-the-art NLP techniques, RDF and SPARQL queries for identifying semantically similar questions. In Chapter 4, we discuss our implementation and the problems involved during the implementation phase. Chapter 5 states the evaluation and is a discussion on the attempt to find out how good is this approach and if this approach can be extended to make it better. In Chapter 6, we provide concluding remarks on the dissertation and an analysis of how successful was this dissertation. It ends with some final remarks, the current limitations and areas for future work.

Chapter 2

Background

In this chapter we discuss the history of question answering systems in general and community question answering sites in particular. The identification of similar questions from the archives of these community question answering sites has been discussed.This draws on to the state of the art approaches to identify semantically similar text and the work done in relevant literature in predicting user's satisfaction depending on recommended questions. We discuss the current techniques used to extract knowledge from the CQA sites and most importantly finding user's intention for querying a CQA site. Quora's graph structure has been briefly discussed followed by a brief discussion on the current challenges in Natural language processing and linked data in the field of question answering.

2.1 Forerunners

Question Answering (QA) Systems or frameworks have evolved, changed and transformed a lot in the recent decades. It has kept pace with the progress of natural language processing. Natural language processing(NLP) progressed quite a lot in the 70s and the 80s, bringing about bigger undertakings in the field of question answering. One such project was the formation of Unix Consultant which was created by Robert Wilensky at U.C. Berkeley in the late 1980s. In 1990s, Boriz Katz along with his associate at Info Lab Group at the MIT Computer Science and Artificial Intelligence Laboratory built a framework called START natural language system. It was the world's first online QA system which has been running non stop since December 1993. The later part of the decade saw a massive surge in the number of search engines and by mid 2000, multiple question answering sites were built based on these search engines.

Various open source text processing projects were undertaken, few of which are Ephyra, YodaQA, PTStemmer, GATE and OpenNLP. In 2003, the Defense Advanced Research Projects Agency (DARPA) financed a Personal Assistant that Learns (PAL) program from which a spinoff named SRI

International, Siri, Inc. was formed, this venture was later acquired by Apple in the year 2010. SIRI was later incorporated in iOS (version 4), an operating system which is manufactured by Apple Inc and used specifically in mobile devices. Around the same time, research associates at IBM were developing Watson [23] to compete with the winners at Jeopardy! - an American game show. IBMs Watson ultimately triumphed over two human champions in the year 2011, a project which started in the year 2007 and opened gateways for further research in the field of deep analytics, natural language understanding and open domain question answering systems.

A few text-based QA systems were developed later, such as QANUS [24]. These QA systems retrieve answers from textual sources and the web. Structured QA frameworks like AquaLog [25] - which is an ontology - driven QA system and they retrieve the answers from pre-defined ontologies and organized data sources.

2.2 Community Question Answering Sites

The very first community question answering site was built by a South Korean company Naver Corporation named Knowledge iN in the year 2002. Answerbag was one such english language CQA site which was started in the year 2003 and later dissolved in the year 2015. However, Yahoo! Answers introduced by Yahoo! in the year 2005 gained much popularity and influenced CQA platforms in becoming institutionalized. Few closed domain CQA sites such as StackOverflow and StackExchange has gained immense popularity as these sites contains rich information about specific domains which is not found elsewhere in the web. CQA sites such as Quora gains active support from it's community members and has become popular within a short period of time. CQA sites have in the recent years integrated with social network sites such as Facebook and has created compatible widgets to allow their users to access topics from other sites. There are various restrictions implemented in CQA sites ranging from inexperienced members not being able to choose best answers to restricting members from commenting on a specific number of questions per day.

2.3 Identifying questions similar to user's question from archives

Usually most of the renowned CQA sites like Quora, Yahoo! Answers maintains a huge archive of solved questions. Every-time a user asks a new question, these CQA sites tries to find this new question coming from the user in their archives in order to avoid delay in response. The concept of searching for a similar question in CQA sites is to some extent similar to passage retrieval in traditional question answering systems. However, it is to be noted that question to question matching is much more difficult and constricted than question to passage matching. There is a significant amount of work done in the areas of passage retrieval, for example by using dependency relations [26] which is a state of the art technique. We have also incorporated dependency relations in our work. Natural language parsing is a key problem to many tasks that requires natural language processing.

The study of finding similar questions is also inspired by the work of finding information in the archives of Frequently Asked Questions(FAQ). Statistical and semantic similarities between questions to rank FAQs has been extensively studied in FAQ Finder [27].In this work [27]the authors used vector space models(VSM) for calculating the similarity and WordNet [28] was used to find a semantic similarity estimate between two questions. We used WordNet to retrieve synonyms for the extracted relations from each question. Lai et al. [29] developed an approach to mine FAQs from the web, however, the archives of CQA are different from the FAQ collections. The major difference between FAQs collection and CQA question archives is in the quality of both the questions and the answers. FAQ collections are usually created and maintained by private(depending on organizations) experts and the quality of questions and answers are therefore much better than CQA archives.

2.4 State of the Art Techniques

Two natural language sentence may mean the same but differ in the way the sentence is formed. For example, ”While pursuing MSc can I start my own blog?” and ”Can I start my own blog while pursuing MSc?” indicates the same meaning but differs in the way both these sentences are worded. The intuition that words with similar meaning will occur in similar context is consistent but there is a limit to how far this idea can be taken [30]. There are various approaches proposed in relevant literature for identifying similar questions, some of them has been briefly described below :

1. **Topic Modelling Approach:** This approach assumes that a particular question and their respective answer (question answer pairs) shares the same topic distribution. Researchers have developed a model to learn the latent topic space (which only emerges during the topic modelling process, therefore called latent) in these question answer pairs. This latent topic space is discovered by the topic models which helps in identifying similar questions and addressing the lexical gap problem [31] [32].

2. **Classic Term Weighting**: Research in the field of Information Retrieval related to combining document models and query models and thereby calculating the similarity based on the weights of the similar texts within documents has been proposed. [33].

3. **User Click Logs:** This approach assumed that if two questions have similar retrieval results or similar click logs then the questions are semantically similar [34] [35].

4. **Translation Models:** Translation models usually defines the mathematical relationship between two or more languages. If two English language sentences ’e1’ and ’e2’ can be trans-

lated to a sentence 'f1' in second language supposedly French, then the two sentences are said to be semantically similar. Phrase Based Translation[36], Word Based Translation [37] and ClickThrough-based translation models [38] are some of the translation models which has been proposed in relevant literature.

5. **Ratio-Based Word Weighting**: A very recent approach by Bae et al.[3] studied on the assumption of similar questions having same or similar categories, a retrieval model was proposed which used ratio-based word weighting for fetching the importance of words in categories and the category importance was estimated using language modelling, translation based language modelling and category based model. Figure 2.1 represents the combination of baseline models employed to propose a ratio-based word weighting model.

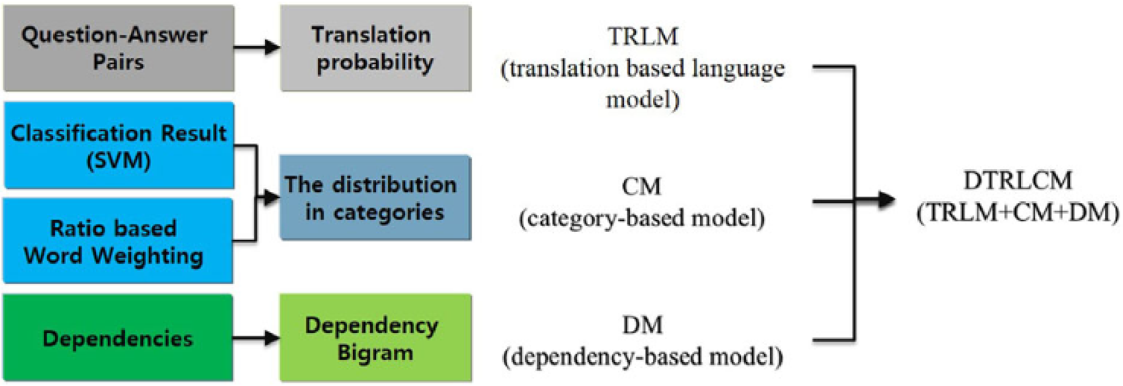

Question-Answer Pairs
Translation probability
TRLM (translation based language model)
Classification Result (SVM)
Ratio based Word Weighting
The distribution in categories
CM (category-based model)
DTRLCM (TRLM+CM+DM)
Dependencies
Dependency Bigram
DM (dependency-based model)

Figure 2.1: Proposed approach for ratio-based word weighting [3]

6. **Deep Learning:** Question-question pair similarity is trained and tested in different variants of neural networks to predict semantically similar questions [39] [40].

2.5 Predicting user's satisfaction depending on recommended questions

A significant amount of work has been done [41] in studying the success of question answering and the dissatisfaction problem of information seekers in CQA. There are mainly two scenarios which keeps an asker unsatisfied and they are unsatisfactory answer and no answer. One approach to increasing user satisfaction is to submit new questions to expert users. Related research [42] defined quantitative measures of expertise to locate expert users. They presented an evidence of validating this quantification as a measure of expertise and used it to find expert users who are best suited to answer a question.

In the field of question answering recommendation there is some relevant research conducted to find the best answers. In [43] a new topic model to simultaneously discover topical distribution of words, categories and users in question answering communities is introduced to find a ranked list of relevant answer. User activity and their related authority information alongwith Latent Drichlet Allocation model is used [44] to find a list of best answerers.

In relevant research[41] it was observed that 78% of the "best answers" were found to be reusable when applied to future similar questions in Yahoo! Answers, however only 48% of the answers were exclusively the best.

This brings us back to our research question of identifying semantically similar questions which can be used as a measure for predicting users satisfaction. Questions which has already been answered can be reused and recommended to the users seeking information thereby enhancing user experience.

2.6 Extracting Knowledge from CQA sites

The ever increasing amount of data in CQA sites indicates a rich knowledge repository which resides in the form of natural language plain text. Extracting this knowledge and populating knowledge bases(KB) could be helpful in enriching knowledge repositories. There is a growing interest in automatically extracting information from these CQA sites and storing them in a structured format using KBs. Inspite of the growing interest, very little work has been done in extracting information from CQA sites for the purpose of storing them in knowledge bases and acting upon them for similar questions retrieval or question answering. Earlier research on fact extraction considering predefined relations between entities in text for supervised learning [45] and an unsupervised approach of extracting string of words between entities in a large corpora to cluster strings to produce relation string [46] has been studied. Another approach which has been used is of bootstrap learning [47] wherein seed instances are used to extract new patterns in an iterative process. This approach, however, suffers from semantic drift as well as low precision. It is manually very costly to label large corpora to create samples for training in the case of supervised learning approaches, unsupervised or weakly supervised approaches are used in case of large datasets. One missing pattern can be detrimental for the entire model and it's corresponding task.

2.7 Finding user's intention for querying a CQA site.

A lot of work has already been done in finding user's intention by categorizing the question into different types. Rose et al. [48] proposed taxonomies with additional subcategories to earlier work of Broder [49] who categorized search queries in three categories namely navigational, informational and transactional. In the area of open domain question answering research, all of the highly rated systems had incorporated question taxonomies in their systems, [50],[51], [52]. Liu et al. [53] made some modifications to accommodate certain specifics of CQA services while following the work of Rose and Levinson (RL) [48]. Figure 2.2 demonstrates the taxonomy proposed by Liu et al. [53]. In this work they have retained the work of Broder's taxonomy [49] at the top levels and proposed a new Social category. The other categories such as Navigational, Informational and Transactional has been defined as it is in Broder's taxonomy, however, the Social category represented questions that didn't intend to get an answer but were just posed to start a conversation with people using CQA services. Along similar lines, Navigational category contained questions which seeks for pointers or URLs of specific websites as their information need. Transactional category contained questions which seeks for specific resources, for example, "What is the best recipe to cook baked beans?". Informational category is segregated in two subcategories *Constant* and *Dynamic*. Constant questions have a fixed set of answers for example,"Which country is the hottest in the world" while the answers for Dynamic questions are not defined e.g. "What is the temperature in Libya?". Dynamic category is further subdivided into three subcategories, *Opinion, Context Dependent* and *Open*. Opinion questions are those asking for opinions from people in CQA sites about what they think about some people, place or thing. "Is studying MSc right after college, worth it?" is an example of Opinion questions. Context-dependent questions are those questions for which the answer varies depending on the context, for example for the question, "What is the temperature in Libya?" the answer changes from minute to minute. Open questions are questions which are seeking for some facts or methods. There can be many different answers for Open questions. "Would you be able to list down all the citations and complete writing your MSc report in a month?" is one such example of open questions. Open questions are those questions which cannot be labelled in opinion or context dependent categories.

Most of the questions in CQA sites are opinion and open questions. The least number of questions are navigation questions as one would use a search engine to find the answer for a navigation question rather than choosing to use a CQA site for this purpose[53].

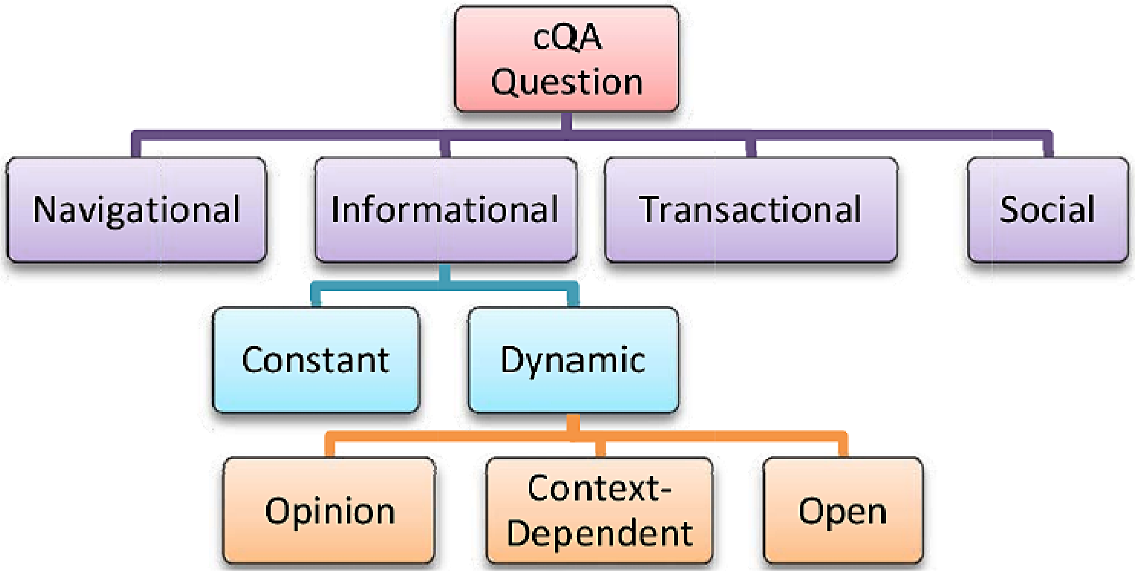


Figure 2.2: CQA Question Type Taxonomy

2.8 Quora's Graph Structure

Since we have used a dataset published by Quora, it is worthwhile to briefly discuss about the internal structure of Quora. The structure of community question answering sites are a combination of questions, questions topics, answers and users. Figure 2.3 from Wang et al. [4] summarizes the relationships between Question, Topics and Users in Quora. Users can follow topics as well as other individuals. Questions in Quora are related to other questions and each question can be tagged using labels. For each question in the system, there are three participants, users who posted the question (Asker), the one who answered the question (Answerer) and the participants who voted on an answer(Voters).

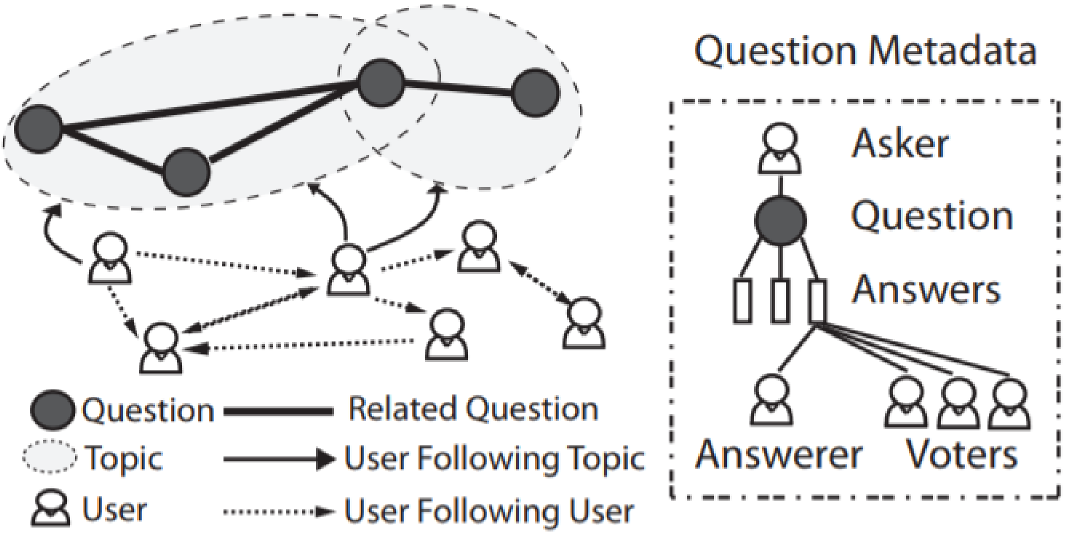


Figure 2.3: Structure of questions, topics and users in Quora [4]

Quora has it's own feature which has an ability to locate questions ”related” to a given question. In this related question graph, nodes represents the questions and the edges represent a similarity measure as determined by Quora. Quora's internal structure is primarily dominated by three graphs.

1. User Topic Graph : Every user in Quora follows a set of topic based on which they receive updates of questions.

2. Social Graph : Quora users can follow each other which gives rise to a social graph. Users receive notifications about the activities about these personalities whom they followed.

3. Question Graph : This graph is a feature used by Quora to map related questions for users to browse them quickly.

These three graphs are believed by Wang et al. [4] to be largely responsible for gaining attention of Quora users. Out of these three graphs the question graph is of primary interest for us as it is an important feature and provides Quora with an ability to locate questions ”related” to a given question. The current production model for solving the problem of identifying similar questions, Quora uses a random forest model with handcrafted features, including the cosine similarity of the average of the word2vec, the number of common words, number of common topics labeled on the question and part of speech tags of the words. [54].

2.9 Challenges in Natural Language Processing and Linked Data

With the growing amount of structured knowledge on the web, the need for providing users with access to this knowledge increases steadily. However, the technical aspects related to data modelling,formulation of vocabularies and query languages is abstracted from the user. The main challenge for question answering over Linked Data is the difficulty to translate user's information need in a format such that it can be evaluated using standard Semantic Web query processing techniques. In case a web user wants to find information in the linked data cloud s/he would face two major obstacles. The first problem that the user will face is in the identification of a relevant dataset which contains the information that s/he is looking for, this is a very big challenge as the amount of datasets available in linked data cloud is huge. Therefore, identifying relevant dataset is not a trivial task. The second problem being the ability to formulate a query e.g. in SPARQL Protocol and RDF Query Language (SPARQL) which is the standard query language for RDF data for accessing the information from the dataset. One cannot assume that a common web user would be familiar with Semantic Web Languages or the available datasets and the structure of the linked data cloud. This is where a question answering system aims to bridge the gap between a user's information need and the structure of the data. An information need expressed in natural language on one side and structured queries to retrieve the information on the other side. Another major challenge in dealing with natural language in general is of ambiguities. A natural language expression can have more than one meaning and this causes problems in mapping to more than one vocabulary element in the target dataset. As the amount of non English speaking web users creating and publishing data in their native languages are increasing, it's important to achieve a common goal that users from all nations irrespective of their language should have access to the same information. Eventhough,in principle, Semantic web is well suited for multilingualism as the URIs are language independent identifiers, a need for labels in order to access these URIs in different language contexts is imperative.

Chapter 3

Design

The previous chapters introduced the current challenges of finding similar questions in CQA sites, the attempts made in relevant literature to find a solution to this problem by introducing various techniques. It also discussed our motivation behind attempting a linked data approach to identify semantically question. This chapter discusses the different categories of linked data applications and respective categorization of our approach. It illustrates the conceptual architecture at an abstract level. It discusses briefly on the different phases of the design and summarizes the effectiveness and limitations of such an approach.

3.1 Overview

According to ”Constructivism”, a term which refers to an idea that learners construct knowledge for themselves in a manner that each learner constructs meaning as s/he learns. In a work on constructivist learning theory [1] a strong assertion is made, ”Constructing meaning is learning; there is no other kind”[55]. Along these lines, any form of new knowledge is learnt on the basis of what is already known by the learners. Emerging technologies such as Linked Data which advocates semantics could be made relevant for the users in this context. Such an application design to dynamically create and maintain knowledge bases is possible through a structured approach such as linked data.

We made two important design choices. Firstly the knowledge of users is captured in a knowledge base in the form of RDF triples which is the atomic unit of information in linked data. Second, the data used to query the knowledge base is also captured in an event when such knowledge is already not present in the knowledge base.

Our current system has four phases: a relationship extraction phase, an open domain ontology creation phase, a query execution phase and a phase for updating knowledge into the knowledge base.

[1] https://www.exploratorium.edu/education/ifi/constructivist-learning

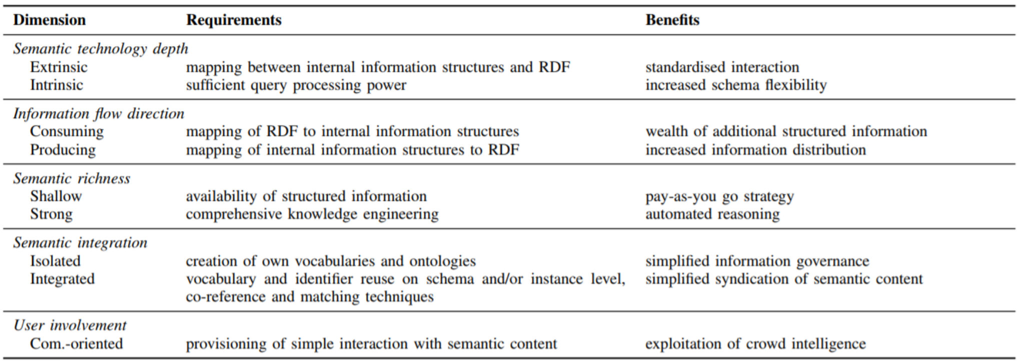
3.2 Categorization of our Linked Data approach

Research related to categorization of linked data application indicates that such applications can be categorised based on dimensions that describe different technical aspects of using and representing Linked Data [5]. Figure 3.1 is a representation of the various dimensions, requirements and their specific benefits based on the categories of Linked Data applications. Applications based on Semantic web technologies if used extrinsically implies that linked data is consumed and processed by Application Program Interface (API) and for storage a relational database is required , while if used intrinsically the internal state of the application is stored in a triple-store rather than in a relational database. A linked data application can consume as well as produce linked data. Our proposed application consumes as well as produces linked data as it employs the RDF data model for internal representation of information.

There is a difference between shallow semantics and strong semantics and the important parameter lies in the kind of relationships between resources. Class properties, memberships and hierarchies can be expressed by RDF and RDF-Schema, however for increasing the richness of the semantics as well as the high level representation formalisms, variants of Web Ontology Language (OWL) is used. In terms of the expressivity of the knowledge representation techniques, the proposed application is rather constrained and mostly in RDF and RDF-Schema space. The OWL features used are mostly confined to class and property definitions. Therefore, semantic richness of the application's information space can be termed as *shallow*.

Applications using Linked Data can also be categorized in two levels *Isolated* and *Integrated*. If an application uses it's own vocabulary which is different from the available vocabularies then such an application is termed as *isolated*. An application which uses, reuses and interlinks other vocabularies extensively is termed as *integrated*. An isolated application can be converted to an integrated application if such an application is published and the vocabularies are used to interlink other dataset. At the present moment our application is in the isolated level as this application has not been published in the Linked Open Data (LOD) cloud.

Exploiting crowd intelligence for enrichment of knowledge base is also attempted by linked data applications. We consider user involvement to be primary in enriching the knowledge base and use this dimension in our proposed proof of concept application. Although an appropriate moderation process is required to prevent the publication of inappropriate material, at the present moment, application logic is used to make modifications in the predefined properties of the knowledge base.



Dimension	Requirements	Benefits
Semantic technology depth		
Extrinsic	mapping between internal information structures and RDF	standardised interaction
Intrinsic	sufficient query processing power	increased schema flexibility
Information flow direction		
Consuming	mapping of RDF to internal information structures	wealth of additional structured information
Producing	mapping of internal information structures to RDF	increased information distribution
Semantic richness		
Shallow	availability of structured information	pay-as-you go strategy
Strong	comprehensive knowledge engineering	automated reasoning
Semantic integration		
Isolated	creation of own vocabularies and ontologies	simplified information governance
Integrated	vocabulary and identifier reuse on schema and/or instance level, co-reference and matching techniques	simplified syndication of semantic content
User involvement		
Com.-oriented	provisioning of simple interaction with semantic content	exploitation of crowd intelligence

Figure 3.1: Categories of semantic web applications [5]

3.3 Conceptual Architecture

According to Soni et al. [56] a conceptual architecture illustrates the important design elements of the system and it also describes the relationship among the design elements. A conceptual architecture is independent of the decisions made during implementation and it emphasizes on the interaction protocols between the design elements [56]. Architectural styles of linked data based application varies and is dependent on the constraints of the application. Our architecture reuses existing libraries and tools as components. A high level description of the components used in order to implement Semantic Web Technologies and Linked Data principles in our proposed application is discussed in the following section.

3.3.1 Description of Components

To discuss the component-based conceptual architecture we primarily introduce the following high-level components. Figure 3.2 represents the components and connectors of the conceptual architecture.

- Data Pre-Processing : This component is used to process data before using it for any further consumption by other components of the system.
- Data-modelling Service : This component provides a mapping from the unstructured data to a graph based data model such as RDF.
- Triple Store : This component provides a persistent storage for RDF.
- Query Engine : This component provides the ability to access the graph based data model by executing queries.

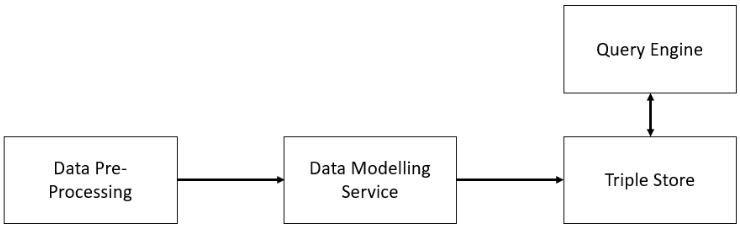


Figure 3.2: Components of conceptual architecture.

The data pre-processing task is responsible for processing the data based on anticipated problems for creating resources. Since the data we use are in natural language we use wrappers to translate the data in the form of semantic triples.We use these triples to form the data model for storing the RDF data in the triple store. We use a query engine to express queries over RDF data for finding a match in the triple store.

3.4 Relationship Extraction

Before persisting data in the triple store we extract the relations from a natural language question using two techniques. Firstly, we adopt the approach of extracting semantically similar questions using treebank parser from the work of Rusu et al. [57] and secondly we use the Open Information Extraction (OpenIE) technique [58] for relationship extraction. Once the triples has been extracted using these two approaches, they are integrated with respect to each question and used to persist data in the triple store.

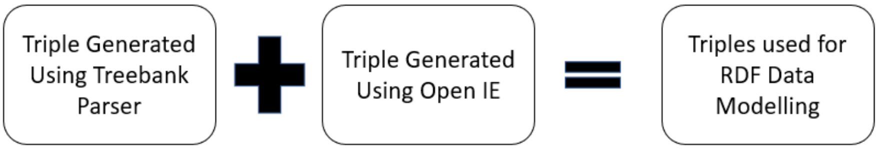

Triple Generated Using Treebank Parser
Triple Generated Using Open IE
Triples used for RDF Data Modelling

Figure 3.3: Integrating triples generated from two approaches

State-of-the-art Natural Language Processing techniques were employed to extract semantic triples from the unstructured text. These triples were modelled to make the data eligible to be stored in a triple store. Figure 3.4 represents the high level flow of data from unstructured data to structured data while using a synonym engine to store the possible variants of a word. Once the relationship from a text are extracted in the form of triples, these words (three of them - subject, predicate and object) are fed into a synonym engine to extract their respective synonyms.

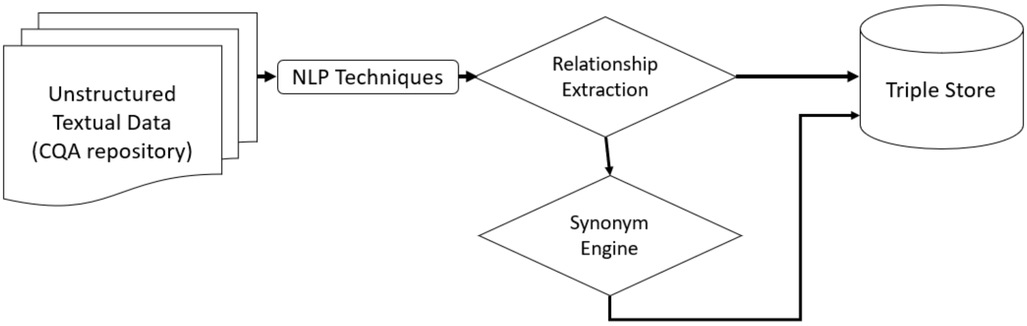


Figure 3.4: Flow Diagram of storing unstructured data in structured format in triple store.

3.5 Synonym Engine

A synonym engine to retrieve the synonyms for all the extracted triples is employed to store semantically similar words as derived from WordNet lexical database. Semantic similarity measure has been researched in the area of information retrieval, text processing which includes text summarization, text categorization and text clustering. All these measures can be primarily grouped in four categories: feature based measures, information content based measures, path-length based measures and hybrid measures [59]. Specific ”senses” of a word in a sentence for which the synonym needs to be extracted is a problem and can be tackled using part-of-speech tagger which helps in determining the context. Figure3.5 presents a sample WordNet hierarchy which demonstrates that the ”senses” of a word can be calculated based on the nodes and the hierarchy. In our approach the triples that are extracted from a natural language sentence are used to find the synonyms with synset types as ’noun’ and ’verbs’. All the synonyms generated are further used to add description to a resource. These synonyms are also used to find a match during query execution when the searched resources are not present in the knowledge base.

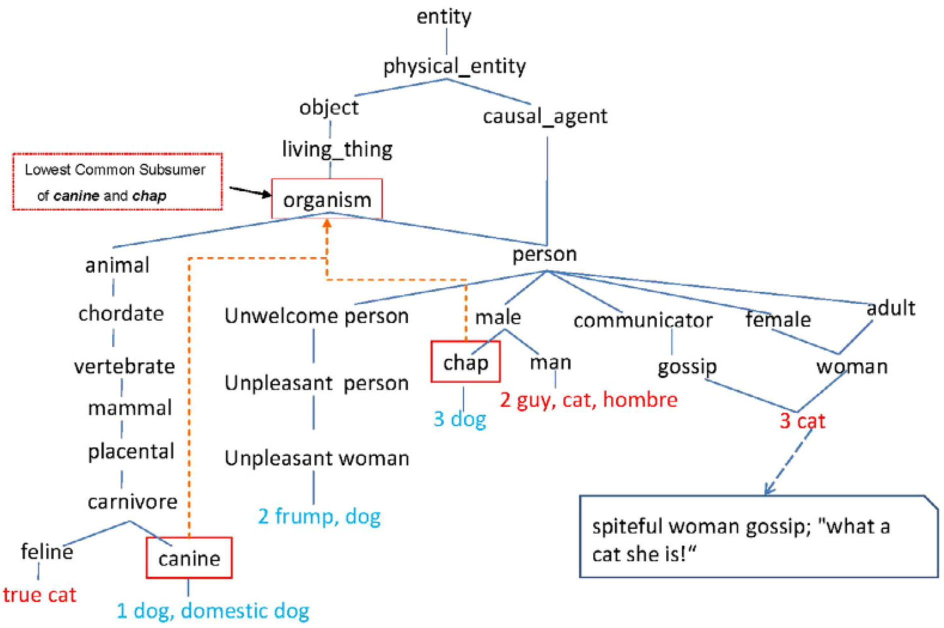


Figure 3.5: A sample WordNet hierarchy [6]

3.6 Open Domain Modelling

In Linked Data ”anybody can say anything about anybody” and everything is structured in triples. We have modelled the universe of discourse for our application in such a manner that we have used only two class definitions ’Subject’ and ’Object’. ’Predicate’ is used to describe the aspects of subject and is primarily used to establish a relationship between Subject and Object. Subject is the ’thing’ or resource that is of interest and Object can be both a ’resource’ or just a ’literal’ such as a value in it’s raw form, for example a word or a number.

In this approach the domain model is a set of RDF triples which includes RDFS and OWL definitions. It is not restricted to any specified domain, instances of resources represents the information items in the RDF graph. Variants of semantic triples generated are used to form knowledge in the triple store and such an approach facilitates more possibilities of representing the same sentence in multiple set of semantic triples which are modelled in RDF graph and used for querying.

In order to perform computations over data to modify resources or to present data to the user, open domain modelling is advantageous when dealing with data of CQA sites, as the questions posted in community question answering sites are diverse and is not always constrained to any specific domain. In our approach we choose such a modelling scheme because of the extensive diversity of questions and uncertainty of questions being posted in CQA forums.

3.7 RDF Triple Store

A Resource Description Framework(RDF) repository is a basis for implementing the identification of semantically similar question system. The RDF repository also referred to as the knowledge base, includes a large number of triples that assists a computer(machine) in recognition and understanding. RDF data in our approach is modeled as a directed, labeled graph G = (V,E) where V is a set of nodes and E is a set of directed edges. In our approach we map the triples generated from a particular question to it's corresponding resources. Each triple is expressed in the form of 'a subject','a predicate' and 'an object'. For example for a given sentence: A student studies computer science the following are the components, a subject "student", a predicate "study", a object "computer science". Using these triples generated from the relationship extraction module a RDF repository has been created. The following are the features of the semantic graph.

- Outgoing links – The node occurred as subjects in the graph.
- Incoming links – The node occurred as objects in the graph.
- Edges of the graph — Represents the established relationship between two nodes of the graph.

We used this approach to formulate a semantic graph to contain the information of natural language in the form of triples.

3.8 Our Approach

Table 3.8 illustrates the comparison of our approach with other state of the art approaches as mentioned in section 2.4. Our proposed approach is a human understandable, machine readable and structured approach.

Approaches	Human Understandable	Machine Readable	Structured
Topic Model	No	Yes	No
Classic-Term Weighting	No	Yes	No
User Click Logs	No	Yes	No
Translation Models	No	Yes	No
Deep Learning	No	Yes	No
Proposed-Approach	Yes	Yes	Yes

Table 3.1: Comparison of Proposed Approach with Other Approaches

Our proposed approach is for consuming as well as creating linked data. In the event, a user asks a natural language question, semantic triples are extracted from the question.These semantic triples are used to construct SPARQL queries to search the triple store for a matched result. The matched result is the output in this case and a validation of the matched result and the user-asked question is performed. In case, the matched result and the user-asked question are syntactically different, the user question along with the associated triples are stored in the triple store. Figure 3.6 represents a high-level flow of the process of recommending semantically similar questions and storing the related knowledge in the specified triple store.

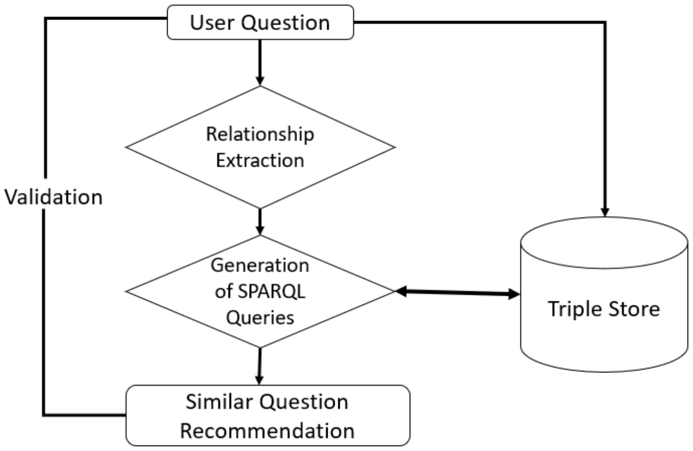


Figure 3.6: Flow Diagram of recommending questions and enhancing the triple store

A high-level workflow of our proposed application is presented in Fig 3.7. The relationship extraction module serves as a service for extracting semantic relations from a question archive to build a knowledge base and the same module is responsible for extracting semantic relation in the event a new question by the user is encountered by the system. After the extraction of the triples, the triples are stored in the triple store and the triples are also processed through a synonym engine for retrieving synonyms, See Figure 3.4 for each entity. The synonyms are stored in the triple store along with the extracted triples. When a new user question is encountered by the system, the relationship extraction module generates the triples which are used to construct SPARQL queries. To find a match in the knowledge base the SPARQL queries are executed on RDF data for retrieving similar questions. If a match is found, the information is presented to the user as a recommended question, in the event a match is not found, the consequent triples generated from the user question is stored in the triple store. If a recommended question and the user asked question is different and agreed to be semantically similar by the user, such information is also updated in the knowledge base for future processing and querying. If both the recommended as well as the user asked question is word to word similar, no update is made in the ontology in this case.

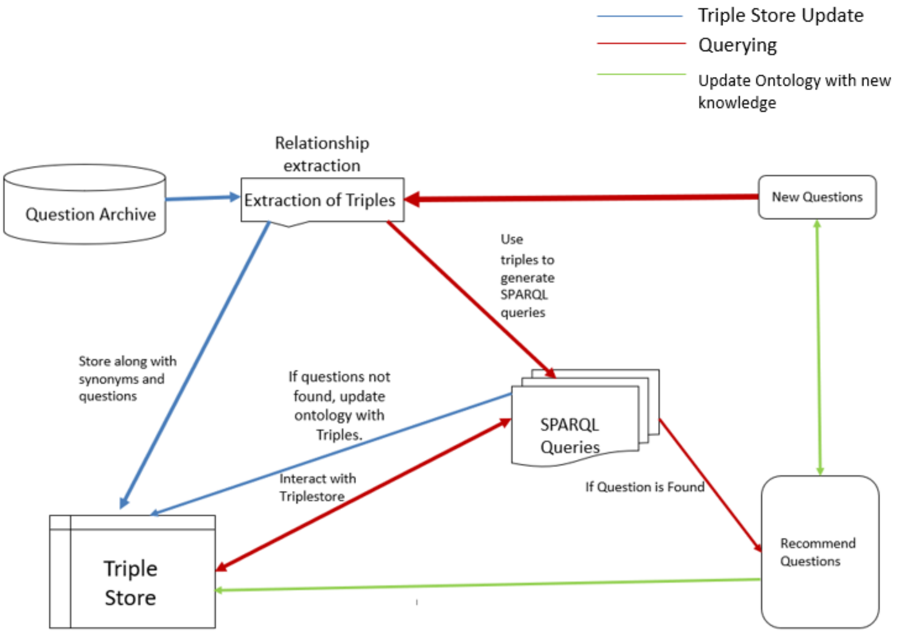

Triple Store Update
Querying
Update Ontology with new knowledge
Relationship extraction
Question Archive
Extraction of Triples
New Questions
Use triples to generate SPARQL queries
Store along with synonyms and questions
If questions not found, update ontology with Triples.
SPARQL Queries
Interact with Triplestore
If Question is Found
Triple Store
Recommend Questions

Figure 3.7: High Level workflow of our proposed approach.

3.9 Summary

We do not claim this to be the best approach to design a model for identifying semantically similar questions. However, at the current moment due to the absence of such a model for finding semantically similar sentences solely using linked data approach which facilitates both human and machine understandable paradigm , this is a proposed idea. This design can be enhanced in multiple ways, such as by using fine-tuned classifiers to rank triples and modelling the system using 5WH questions (What, Where, When, Why, Who and How) as different classes to hold related semantics and questions which can be answered with yes/no as different classes. By using named graphs to store information of specific subjects, which would enable in executing more specific SPARQL queries. Additionally other common variants of questions needs to be appropriately classified into different classes. Our research question is to what extent can natural language questions be structured into RDF graphs and with the extraction of semantic triples and usage of synonym engine to get related words it is evident that natural language can be structured in RDF graph which can then be queried using SPARQL. At the present moment it has a small but significant impact on this concept of structuring unstructured text with the help of semantic triples.

Chapter 4

Implementation

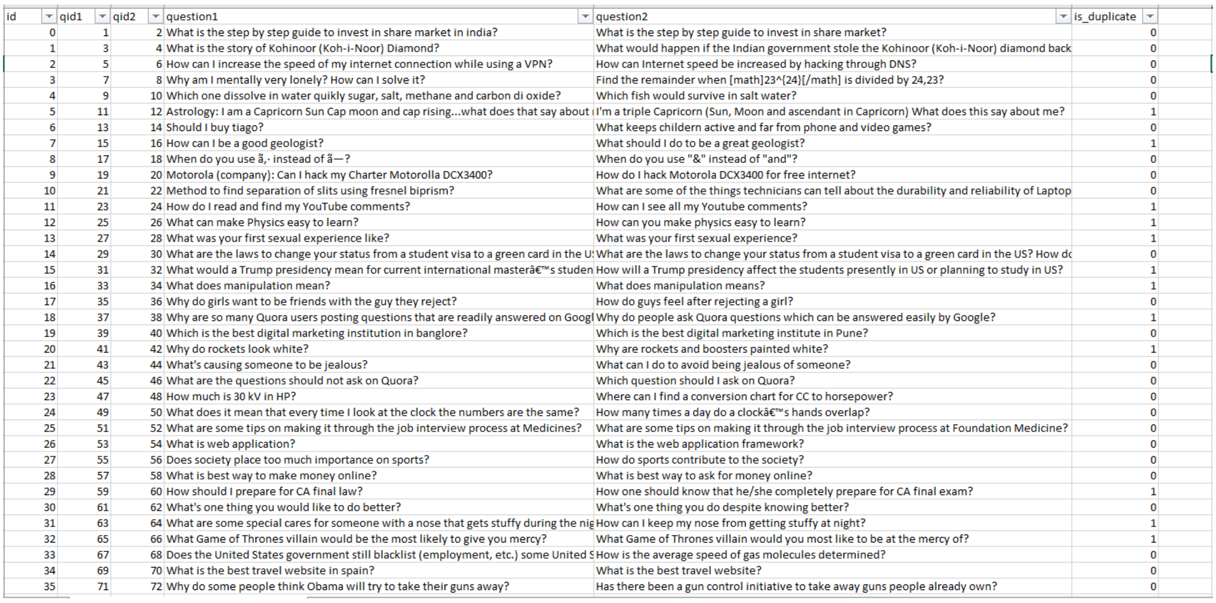
In the previous chapter we discussed the high level design of the application. In this chapter we will be discussing the implementation in details. We implemented our proposed approach as a JAVA application . This chapter discusses on the dataset used, the input and output considerations, how the data was pre-processed, the relationship extraction phase, mapping to triple store and finally interacting with the triple store. We finally summarize this chapter mentioning the accomplishments and the limitations of this approach.

4.1 Dataset

We run our experiment on Quora Question Pairs dataset [60]. This dataset is primarily related to the problem of identifying duplicate question. Identification of duplicate question has been stated as one of the challenges that arise in building a scalable knowledge-sharing platform. The dataset consists of 404,301 lines of potential pairs of duplicate questions. Each line contains IDs of the question, full text of each question and a binary value representing whether the line contains a duplicate pair of questions. Currently, Quora uses a Random Forest model to identify duplicate questions [60]. The data is in the following format:

- id - unique id of the question pair.
- qid1, qid2 - unique ids of question 1 and question 2.
- question 1, question 2 - the complete text of each question
- is_duplicate - if question 1 and question 2 are semantically similar then the value is '1', otherwise the value is stated as '0'.

A brief snapshot of the dataset is shown in Figure4.1. Labelling of the question being semantically similar is inherently subjective as it is annotated by human experts.



id	qid1	qid2	question1	question2	is_duplicate
0	1	2	What is the step by step guide to invest in share market in india?	What is the step by step guide to invest in share market?	0
1	3	4	What is the story of Kohinoor (Koh-i-Noor) Diamond?	What would happen if the Indian government stole the Kohinoor (Koh-i-Noor) diamond back	0
2	5	6	How can I increase the speed of my internet connection while using a VPN?	How can Internet speed be increased by hacking through DNS?	0
3	7	8	Why am I mentally very lonely? How can I solve it?	Find the remainder when [math]23^{24}[/math] is divided by 24,23?	0
4	9	10	Which one dissolve in water quikly sugar, salt, methane and carbon di oxide?	Which fish would survive in salt water?	0
5	11	12	Astrology: I am a Capricorn Sun Cap moon and cap rising...what does that say about	I'm a triple Capricorn (Sun, Moon and ascendant in Capricorn) What does this say about me?	1
6	13	14	Should I buy tiago?	What keeps childern active and far from phone and video games?	0
7	15	16	How can I be a good geologist?	What should I do to be a great geologist?	1
8	17	18	When do you use ã,· instead of ã—?	When do you use "&" instead of "and"?	0
9	19	20	Motorola (company): Can I hack my Charter Motorolla DCX3400?	How do I hack Motorola DCX3400 for free internet?	0
10	21	22	Method to find separation of slits using fresnel biprism?	What are some of the things technicians can tell about the durability and reliability of Laptop	0
11	23	24	How do I read and find my YouTube comments?	How can I see all my Youtube comments?	1
12	25	26	What can make Physics easy to learn?	How can you make physics easy to learn?	1
13	27	28	What was your first sexual experience like?	What was your first sexual experience?	1
14	29	30	What are the laws to change your status from a student visa to a green card in the U	What are the laws to change your status from a student visa to a green card in the US? How d	0
15	31	32	What would a Trump presidency mean for current international masterâ€™s studen	How will a Trump presidency affect the students presently in US or planning to study in US?	1
16	33	34	What does manipulation mean?	What does manipulation means?	1
17	35	36	Why do girls want to be friends with the guy they reject?	How do guys feel after rejecting a girl?	0
18	37	38	Why are so many Quora users posting questions that are readily answered on Googl	Why do people ask Quora questions which can be answered easily by Google?	1
19	39	40	Which is the best digital marketing institution in banglore?	Which is the best digital marketing institute in Pune?	0
20	41	42	Why do rockets look white?	Why are rockets and boosters painted white?	1
21	43	44	What's causing someone to be jealous?	What can I do to avoid being jealous of someone?	0
22	45	46	What are the questions should not ask on Quora?	Which question should I ask on Quora?	0
23	47	48	How much is 30 kV in HP?	Where can I find a conversion chart for CC to horsepower?	0
24	49	50	What does it mean that every time I look at the clock the numbers are the same?	How many times a day do a clockâ€™s hands overlap?	0
25	51	52	What are some tips on making it through the job interview process at Medicines?	What are some tips on making it through the job interview process at Foundation Medicine?	0
26	53	54	What is web application?	What is the web application framework?	0
27	55	56	Does society place too much importance on sports?	How do sports contribute to the society?	0
28	57	58	What is best way to make money online?	What is best way to ask for money online?	0
29	59	60	How should I prepare for CA final law?	How one should know that he/she completely prepare for CA final exam?	1
30	61	62	What's one thing you would like to do better?	What's one thing you do despite knowing better?	0
31	63	64	What are some special cares for someone with a nose that gets stuffy during the nig	How can I keep my nose from getting stuffy at night?	1
32	65	66	What Game of Thrones villain would be the most likely to give you mercy?	What Game of Thrones villain would you most like to be at the mercy of?	1
33	67	68	Does the United States government still blacklist (employment, etc.) some United S	How is the average speed of gas molecules determined?	0
34	69	70	What is the best travel website in spain?	What is the best travel website?	0
35	71	72	Why do some people think Obama will try to take their guns away?	Has there been a gun control initiative to take away guns people already own?	0

Figure 4.1: First few lines of the raw dataset

The plot representing the number of unique questions versus the number of repeated questions in Fig4.2 clearly indicates that the dataset contains more number of unique questions. This helps us in our research as more number of unique questions create more knowledge for our knowledge base.

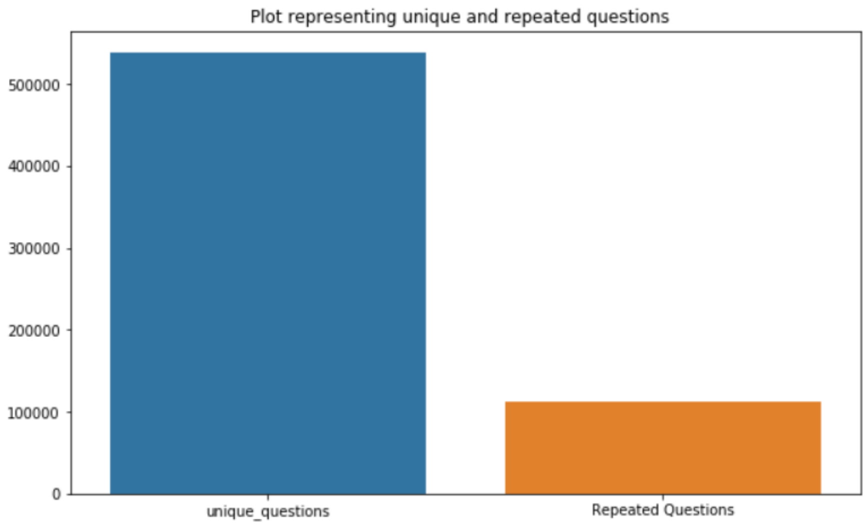


Figure 4.2: Number of unique questions versus repeated questions

4.2 Input

We select a set of 20,000 questions from the Quora Question Pairs Dataset in an iterative manner to populate the knowledge base.These questions are used for further processing which encompasses triples generation and storage in triple store.

For the entire system's framework we expect a question in natural language from a user. This question is then further processed to extract relationship information in the form of triples to interact with the knowledge base.

4.3 Output

For a given natural language question, after the triples extraction phase, we construct SPARQL queries which interact with the knowledge base to find a match based on the triples generated from the question. These matched questions from the knowledge-base are then presented to the user.

4.4 Data Preprocessing

For parsing descriptions and elements of information that is to be used for extraction of semantic relations. The following tasks are applied.

According to the work by Fossati et al. [61] word window and syntactic pattern are taken into consideration where a sentence below the word count of 5 is not taken into account and sentences which lack the basic pattern of Noun-Verb-Noun tags is not considered for relationship extraction. The special characters in a natural language sentence are replaced by natural language words for example, '+' has been replaced with 'plus'. This action has been performed so as to avoid creation of malformed Uniform Resource Identifiers (URIs).

4.5 Relationship Extraction

The main advantages of triple extraction from natural language sentences are manifold. Triple extraction provides us with a compact and simple representation of information contained in the sentence. It avoids the complexity of a full parse on a natural language sentence. The main reason for using triple extraction in our approach is because the triples contain semantic information which could be used to build a knowledge base as well as use the triples to query the knowledge base.

There are essentially two approaches for triple extraction which we have used in our implementation. Extraction using parse tree of a sentence using heuristic rules such as OpenNLP - Treebank ParseTree which has been used in numerous study for entity extraction as well as recommendation systems [62] and the other approach is by Open Information extraction. To organize words in groups which holds grammatical sense parse tree is required. An example of a pictorial representation of a parse tree for a specified question ”Who is the best tennis player in the world” as generated by Stanford Parser is shown in Figure 4.3. The meaning of all the tags are present in the Appendix section. The relevant tagging and the parse representation is shown in Figure 4.4

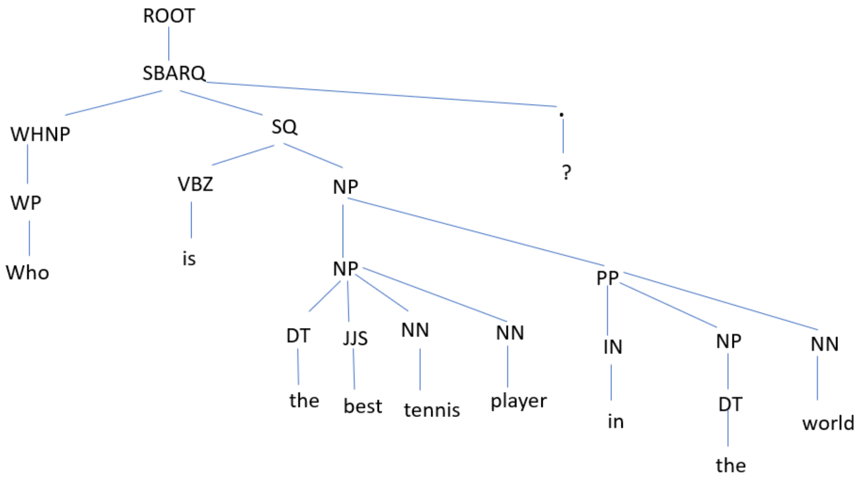
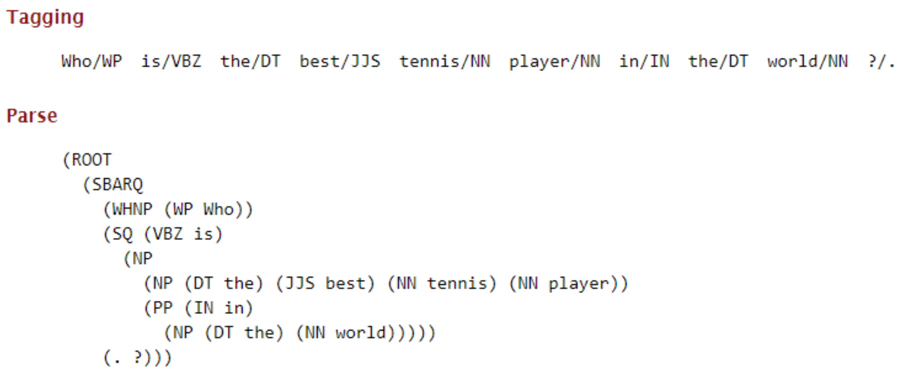


Figure 4.3: Parse tree using Stanford Parser [2]for a question *”Who is the best tennis player in the world?”*



Tagging

```
Who/WP  is/VBZ  the/DT  best/JJS  tennis/NN  player/NN  in/IN  the/DT  world/NN  ?/.
```

Parse

```
(ROOT
  (SBARQ
    (WHNP (WP Who))
    (SQ (VBZ is)
      (NP
        (NP (DT the) (JJS best) (NN tennis) (NN player))
        (PP (IN in)
          (NP (DT the) (NN world)))))
    (. ?)))
```

Figure 4.4: Tags and parse representation using Stanford Parser [4]for a question *"Who is the best tennis player in the world?"*

There are various feature which depends on the triple candidates such as length of the sentences,number of words, context of subject, predicate and object, distance between subject, predicate and object, number of links from subject, predicate and object, depth, diameter, siblings, uncles, cousins, path to root, part of speech tagging which are used to determine the triples that are closely relevant to a given natural language sentence.

In the relationship extraction stage the system first parses a natural language question to a set of triples. Referring to the work of Russu et al. [57] triple extraction using treebank parsers has been adopted in our work. Along with this work, we have used Stanford's Open Domain Information Extraction (OpenIE) which is a part of Stanford CoreNLP. These two methods are used for extraction of relation tuples. The major advantage of OpenIE over other open information extraction systems is that the schema for these relations which are generated need not be specified in advance. Given a natural language sentence, pre-processing of the sentence in various linguistically motivated ways are performed to produce coherent clauses. These clauses are then used to segment OpenIE triples [58]. Open IE triples has been used in a number of applications for structured relations [63] and entailment relations [64] which are critical to many natural language processing applications for semantic parsing and question answering.

4.5.1 Triples extraction using Treebank Parsers

While using a treebank parser a sentence (S) is represented as a parse tree with S being the root of the tree having three main children, a noun phrase (NP), a verb phrase (VP) and a full stop (.) [57].

In order to find the subject of the sentence the subtree of NP is searched using breadth first search. The very first child of NP which is a noun is selected. In the following subtrees nouns can be found.

[2] https://nlp.stanford.edu/software/lex-parser.shtml
[4] https://nlp.stanford.edu/software/lex-parser.shtml

The tags which starts with 'NN' in the subtree of 'NP' is considered for finding the subject in the parse tree representation of the natural language sentence.

Subtree	Type of Nouns
NN	noun,common,singular,mass
NNP	noun, proper, singular
NNPS	noun, proper, plural
NNS	noun, common, plural

Table 4.1: Selecting Subjects from subtrees

For finding the predicate of the sentence, a search is performed in the VP subtree and the deepest verb descendent is considered to be the predicate. The predicates are found in the following subtrees.

Subtree	Type of Verbs
VB	verb, base form
VBD	verb, past tense
VBG	verb, present participle or gerund
VBN	verb, past participle
VBP	verb,present tense, not 3rd person singular
VBZ	verb, present tense, 3rd person singular

Table 4.2: Selecting Predicates from subtrees

Finally for objects, three different subtrees are searched, all the cousins of the VP subtree which contains the predicate. Subtrees which are ADJP(Adjective Phrase), PP(prepositional phrase) and NP. In NP and PP the first occurring noun is searched for while an adjective is searched in the following subtrees of ADJP.

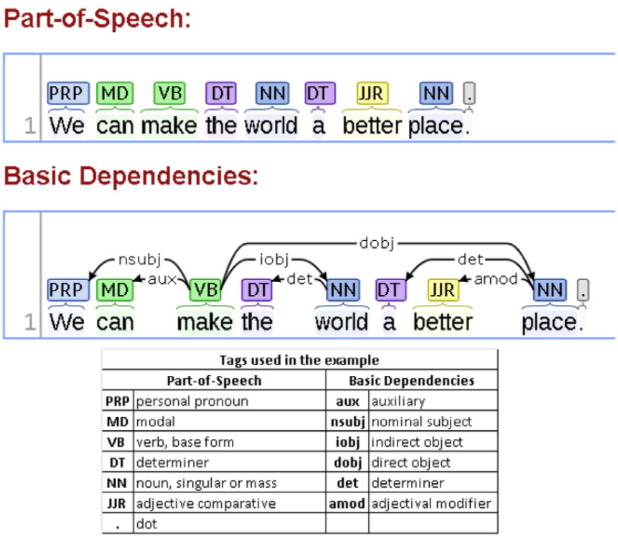
Subtree	Type of Nouns
JJ	adjective or numeral, ordinal
JJR	adjective,comparative
JJS	adjective, superlative

Table 4.3: Selecting Objects from subtrees

Natural Language Toolkit(NLTK) version 3.0 has been leveraged for the purpose of using Stanford Model and Stanford Parser which has been used to carry out the task for generating the subtree and based on the labels of the subtrees the segregation of the subject, predicate and object has been executed. The results for each of the questions in the dataset has been stored in a comma-seperated values(CSV) file for further processing and integration with the triples obtained from the open information extraction approach. The triples generated by this approach is only used during the formulation of the knowledge base and not during the interaction with the triple store, when in response to a user question a set of triples are generated by OpenIE approach which are used to find a similar question.

4.5.2 Triples extraction using Open Information Extraction

Open Information Extraction (IE) triples have been used in a number of applications for entailment graph learning [64]. Stanford Open IE is motivated from the work of finding short entailment from long sentences as described in the work of [65]. A set of clauses which can be considered on it's own as syntactically and semantically correct and is entailed by the original sentence is produced. One very important project which uses Open Information Extraction is the Never Ending Language Learning(NELL) project which is in effect from the year 2010 and since then is processing the web for evolving a knowledge base of facts and relations [66]. Figure 4.5 demonstrates two example of Part-of-Speech(POS) tagging and basic dependencies using Stanford CoreNLP.


Part-of-Speech:
PRP MD VB DT NN DT JJR NN .
1 We can make the world a better place.
Basic Dependencies:
nsubj
aux
iobj
dobj
det
det
amod
PRP MD VB DT NN DT JJR NN .
1 We can make the world a better place.

Tags used in the example			
Part-of-Speech		Basic Dependencies	
PRP	personal pronoun	aux	auxiliary
MD	modal	nsubj	nominal subject
VB	verb, base form	iobj	indirect object
DT	determiner	dobj	direct object
NN	noun, singular or mass	det	determiner
JJR	adjective comparative	amod	adjectival modifier
.	dot		

Figure 4.5: Part of speech and basic dependency generated by Stanford CoreNLP [7]

For different NLP applications including question answering Open IE forms a part of the pipeline. Figure 4.6 represents a general pipeline for Open Information task.

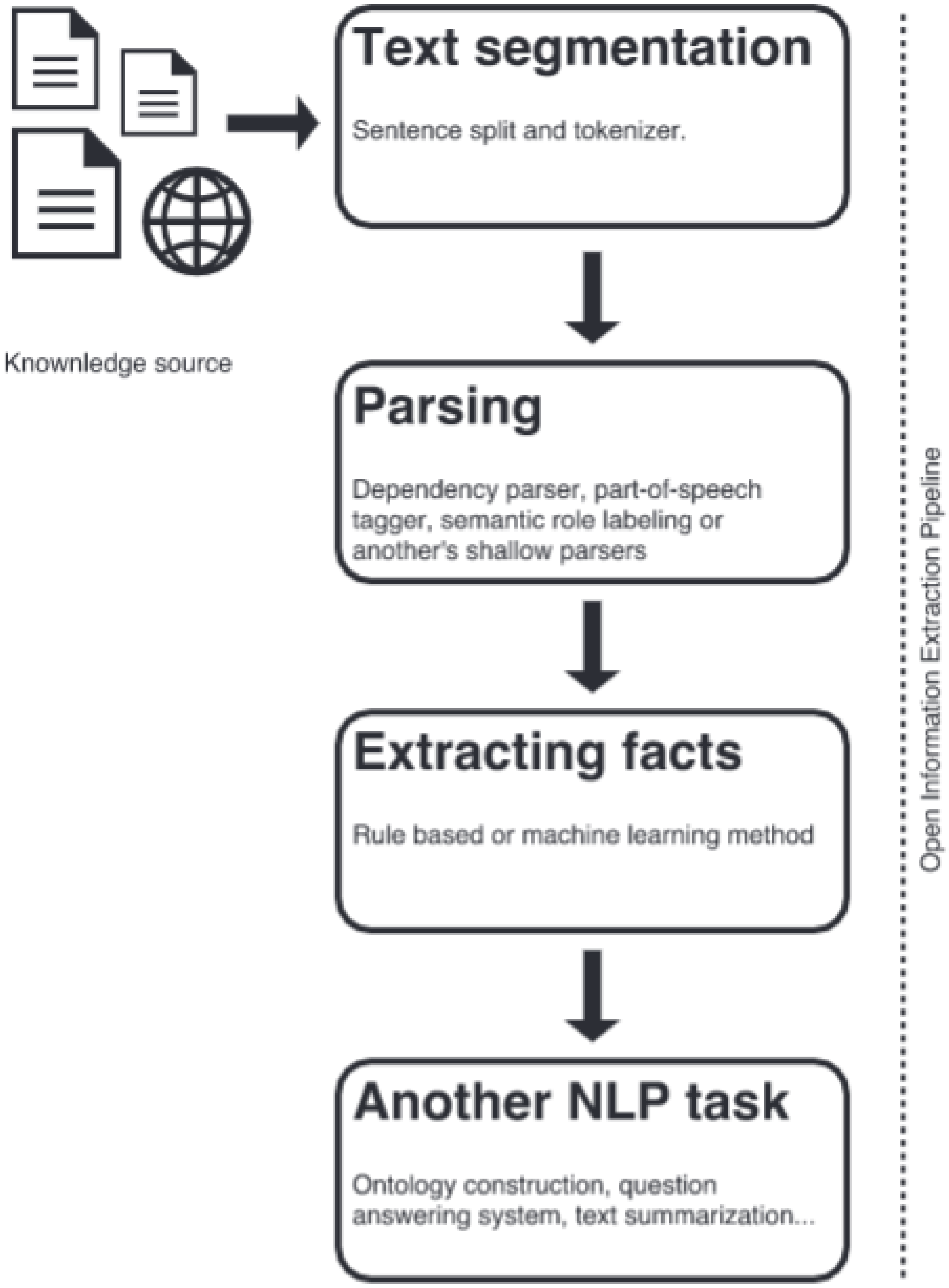
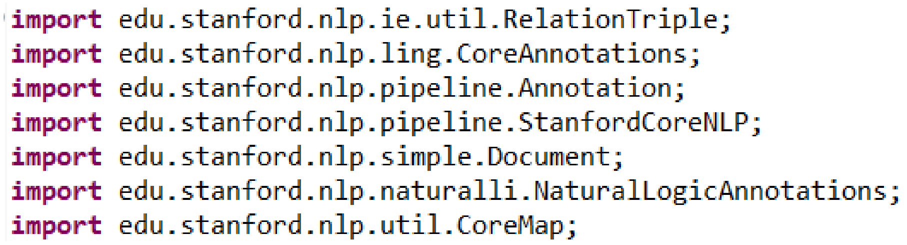

Knownledge source
Text segmentation
Sentence split and tokenizer.
Parsing
Dependency parser, part-of-speech tagger, semantic role labeling or another's shallow parsers
Extracting facts
Rule based or machine learning method
Another NLP task
Ontology construction, question answering system, text summarization...
Open Information Extraction Pipeline

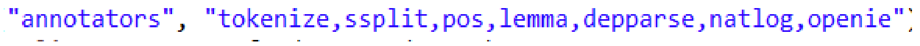
Figure 4.6: Pipeline of Open Information Extraction [7]

In our implementation we used Stanford CoreNLP which is written in Java. The CoreNLP pipeline has been used in our implementation which has been inspired from the work of Manning et al. [67]. It has a lightweight framework which uses Java objects. Figure 4.7 presents the packages that were imported to facilitate the generation of relational triples. Annotators listed in Figure 4.8 are used in our implementation to create a Stanford CoreNLP pipeline. We have used the 3.9.2 version of stanford-corenlp to execute this task. In this process a CoreNLP document object is created which holds the list of sentences and every sentence is iterated over to generate relational triples.



```java
import edu.stanford.nlp.ie.util.RelationTriple;
import edu.stanford.nlp.ling.CoreAnnotations;
import edu.stanford.nlp.pipeline.Annotation;
import edu.stanford.nlp.pipeline.StanfordCoreNLP;
import edu.stanford.nlp.simple.Document;
import edu.stanford.nlp.naturalli.NaturalLogicAnnotations;
import edu.stanford.nlp.util.CoreMap;
```

Figure 4.7: Packages imported for the task of extracting triples by Stanford CoreNLP OpenIE approach.



```
"annotators", "tokenize,ssplit,pos,lemma,depparse,natlog,openie")
```

Figure 4.8: Annotators used to set property for creating Stanford CoreNLP pipeline

4.6 RDF Data Model - Triple Store

Each entity has an associated rdfs:label which we use to store the questions as human-readable label for each entity. Figure 4.9 represents an example of the RDF graph for three independent questions having the same subject. In the representation, *base:Richard_Muller* is the subject which is associated by two predicates *base:be* and *base:do_think_of* to three questions and four objects. The rdfs:label associated with each subject is a question from which the subject was extracted. Similiarly, for each predicate and object rdfs:label is used to specifically store a related question in the form of human-readable label.

4.6.1 WordNet

WordNet is widely used for NLP tasks and serves as a lexical database in which the data is persisted in a related fashion as a semantic network. We use the rdfs:comment to tag the related words for each entity. Along with the questions shown in Figure4.10 the rdfs:comment is used to store the related words of *base:do_think_of* as fetched from JAVA API for WordNet Searching (JAWS) which provides

JAVA applications with the ability to retrieve data from WordNet database. JAWS was developed by Brett Spell as a student project in Southern Methodist University [5]. This API is compatible with both 2.1 and 3.0 versions of the WordNet database.It can be used with Java 1.4 and later versions. In our implementation we have used the 2.1 version of WordNet database and used this API to generate similar word forms and stored them in rdfs:comment for each entity.

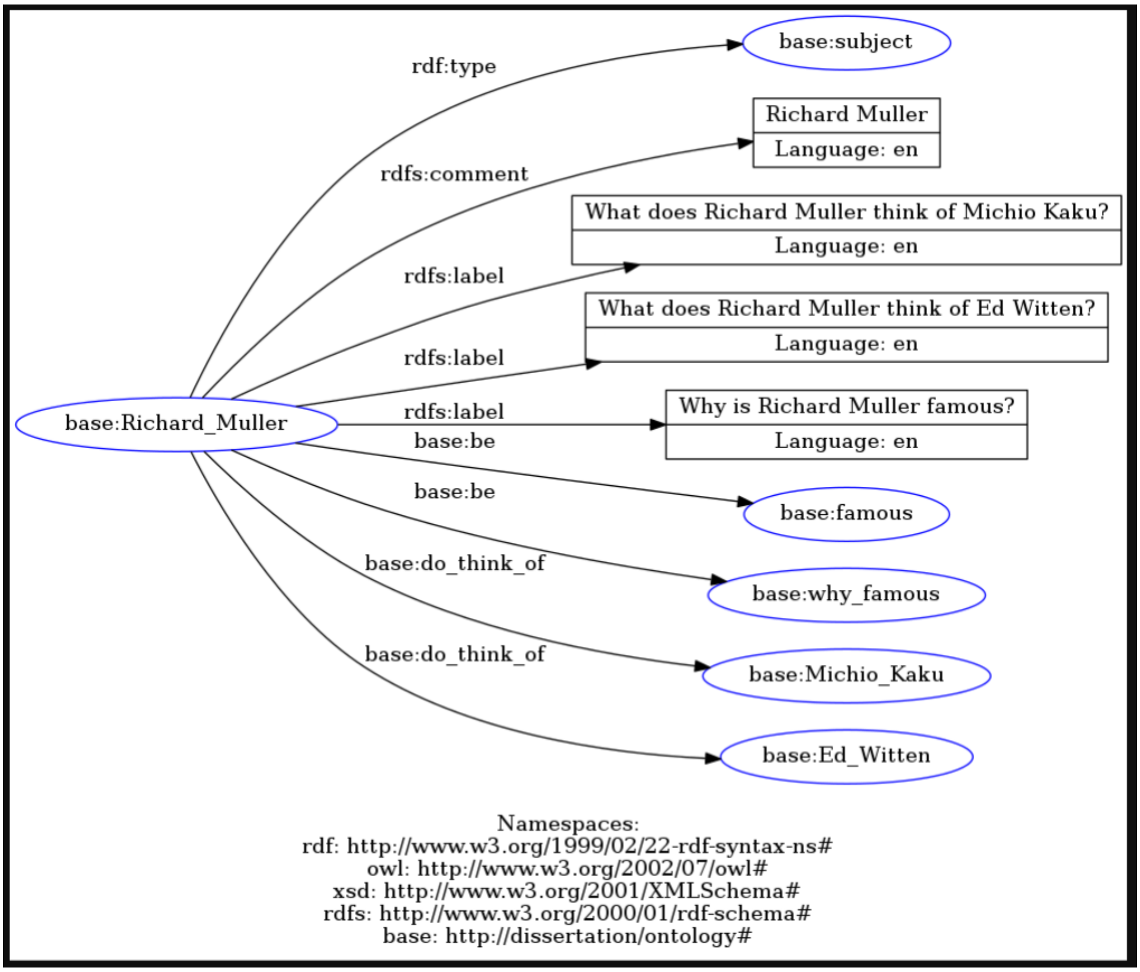

base:Richard_Muller
rdf:type
base:subject
rdfs:comment
Richard Muller
Language: en
rdfs:label
What does Richard Muller think of Michio Kaku?
Language: en
rdfs:label
What does Richard Muller think of Ed Witten?
Language: en
rdfs:label
Why is Richard Muller famous?
Language: en
base:be
base:be
base:famous
base:why_famous
base:do_think_of
base:Michio_Kaku
base:do_think_of
base:Ed_Witten
Namespaces:
rdf: http://www.w3.org/1999/02/22-rdf-syntax-ns#
owl: http://www.w3.org/2002/07/owl#
xsd: http://www.w3.org/2001/XMLSchema#
rdfs: http://www.w3.org/2000/01/rdf-schema#
base: http://dissertation/ontology#

Figure 4.9: RDF graph representation of a question stored in triple store [6]

[5] https://archive.org/stream/PracticalArtificialIntelligenceProgrammingWithJava/JavaAI3rd_djvu.txt

[6] http://www.ldf.fi/service/rdf-grapher

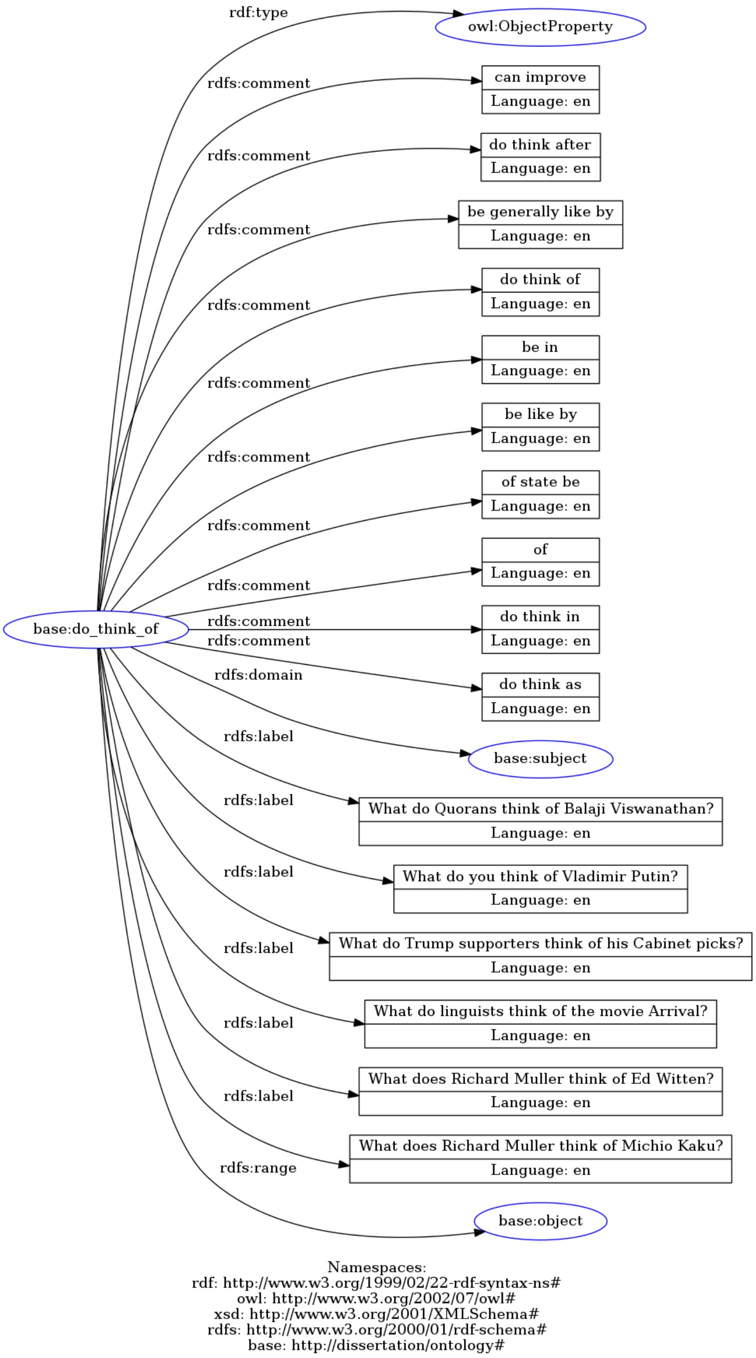


Figure 4.10: Predicate associated to different questions [7]

[7] http://www.ldf.fi/service/rdf-grapher

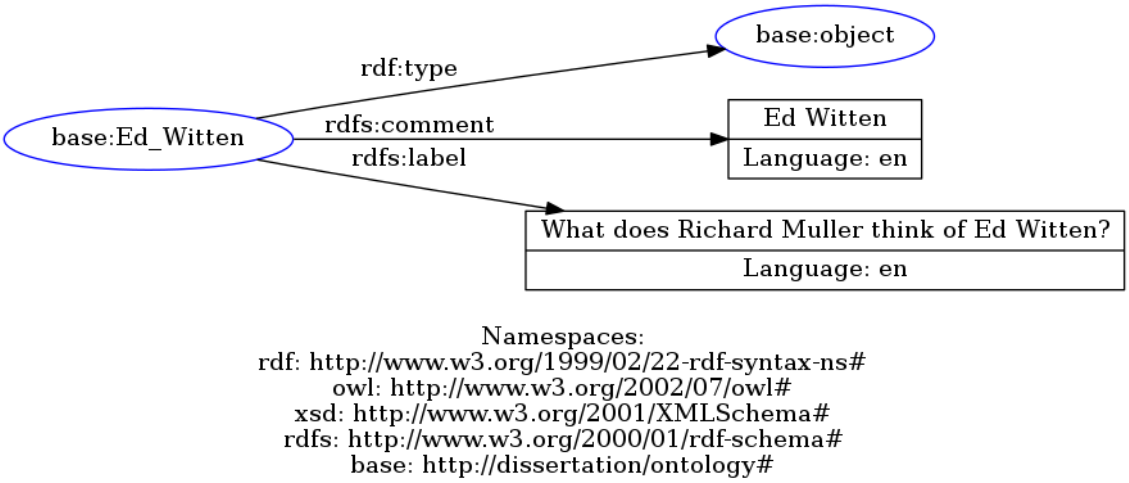

base:object
rdf:type
base:Ed_Witten
rdfs:comment
rdfs:label
Ed Witten
Language: en
What does Richard Muller think of Ed Witten?
Language: en
Namespaces:
rdf: http://www.w3.org/1999/02/22-rdf-syntax-ns#
owl: http://www.w3.org/2002/07/owl#
xsd: http://www.w3.org/2001/XMLSchema#
rdfs: http://www.w3.org/2000/01/rdf-schema#
base: http://dissertation/ontology#

Figure 4.11: Object associated to a particular question: ”What does Richard Muller think of Ed Witten?” [9]

The objects as seen in Figure 4.12 **base:Ed_Witten** and in Figure 4.12 **base:Michio_Kaku** is annotated with the natural language question from which it was extracted using the rdfs:label tag.

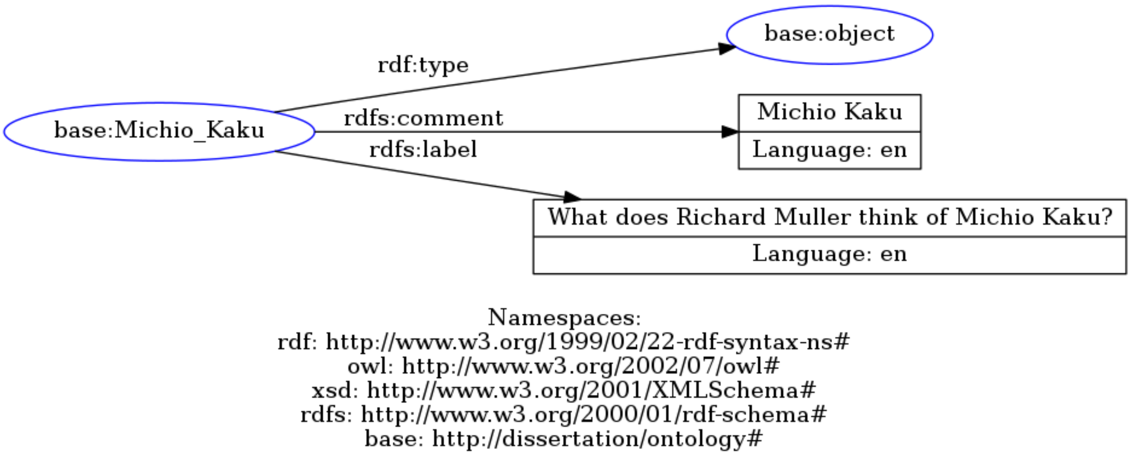

base:object
rdf:type
base:Michio_Kaku
rdfs:comment
rdfs:label
Michio Kaku
Language: en
What does Richard Muller think of Michio Kaku?
Language: en
Namespaces:
rdf: http://www.w3.org/1999/02/22-rdf-syntax-ns#
owl: http://www.w3.org/2002/07/owl#
xsd: http://www.w3.org/2001/XMLSchema#
rdfs: http://www.w3.org/2000/01/rdf-schema#
base: http://dissertation/ontology#

Figure 4.12: Object associated to a particular question:”What does Richard Muller think of Michio Kaku?” [11]

[9] http://www.ldf.fi/service/rdf-grapher

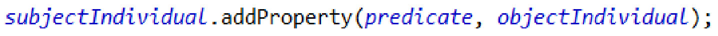
[11] http://www.ldf.fi/service/rdf-grapher

4.6.2 Apache Jena

Apache Jena[12] is a free open source Java framework for developing semantic web and Linked Data applications. In our implementation we have used the 3.9.0 version of Jena for processing RDF data, building knowledge base and formulating queries over RDF data. We use the OntModel interface to create an enhanced view of the Jena model to contain the ontology data.

4.6.3 Model

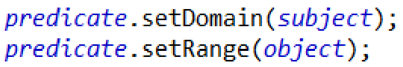
The OntClass interface which represents an ontology node representing a class description has been used in our implementation. The two classes in our implementation are the Subject and the Object and we define the Predicate as an ObjectProperty which is an interface for encapsulating properties whose range values are restricted to individuals. Figure 4.13 shown below is a representation of dynamically generating the RDF data model.



```
subjectIndividual.addProperty(predicate, objectIndividual);
```

Figure 4.13: Using Jena to form RDF data model

Figure 4.14 represents the domain and range values for the predicate. The setDomain function is for asserting a given resource to represent the class of individuals that would form the domain of the property. Similarly, the setRange function is for asserting a given resource to represent the class of individuals that would form the range of the property. In our case, the domain for predicate is set to be the subject and the range is set to be the object.



```
predicate.setDomain(subject);
predicate.setRange(object);
```

Figure 4.14: Definition of the domain and range for predicate

4.7 Interacting with Triple Store

Finding an answer to a query executed on a RDF dataset is a pattern matching problem. Triples which match a particular set of graph patterns is retrieved as a result. We have used TDB which is a component of Jena for RDF storage and query and is used as a RDF store in a single machine. In the event that an user asks a natural language question the relationship extraction module extracts the triples from the question and constructs SPARQL queries using the triples. There are five levels of matches which are performed to find a question based on the triples which are generated. In case the

[12] https://jena.apache.org/getting_started/

question asked by the user is 'new', the definition of 'new' here is a question which is not already present in the knowledge base, the question is updated in the knowledge base along with the extracted triples.

There are five levels of search which is made on the knowledge base to retrieve a similar questions. The following are the descriptions of the levels.

- ***Level 1:*** At this level the SPARQL query searches for the URIs of the Subject, Predicate and Object having a match for each triple that is generated from the unstructured question. In case there are no match found at this level then the application proceeds to the next level of search; Level 2. If a match is found, the system stores the required information for presentation and moves on to the next set of triples generation and tries to find a match at Level 1.

- ***Level 2:*** At this level the SPARQL query searches for the URIs of the Subject and Predicate with a search on the synonym set of the Objects.In case there are no match found at this level then the application proceeds to the next level of search; Level 3. If a match is found, the system stores the required information for presentation and moves on to the next set of triples generated by the system and tries to find a match at Level 1., the search begins at Level 3.

- ***Level 3:*** At this level the SPARQL query searches for the URIs of the Subject and Object with a search on the synonym set of the Predicate. In case there are no match found at this level then the application proceeds to the next level of search; Level 4.If a match is found, the system stores the required information for presentation and moves on to the next set of triples generated by the system and tries to find a match at Level 1.

- ***Level 4:***At this level the SPARQL query searches for the URIs of the Predicate and Object with a search on the synonym set of the Subject. In case there are no match found at this level then the application proceeds to the next level of search; Level 5. If a match is found, the system stores the required information for presentation and moves on to the next set of triples generated by the system and tries to find a match at Level 1.

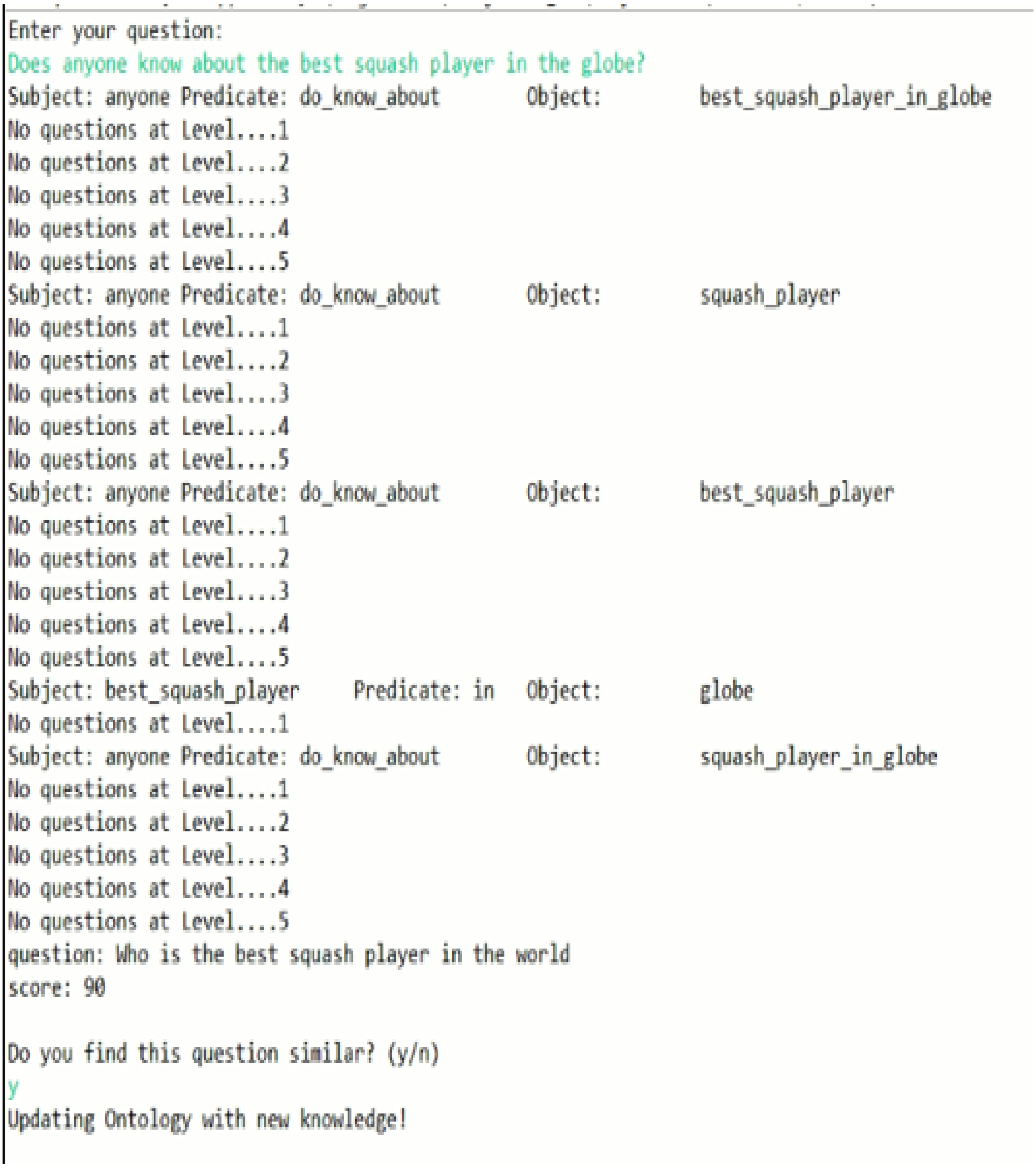
- ***Level 5:*** At this level the SPARQL query searches in the synonym set of the Subjects, Predicate and Object . In case when a match is found, the system stores the required information for presentation . In case there are no matches at this level, there are two tasks which executes, if the system is able to generate another set of triples from the sentence, the whole process of search commences from Level 1. If, the system is unable to generate anymore triples and there are no matches found at Level 5, the triples along with the question is stored in the knowledge base.

Once all the levels of searches for all the generated triples are concluded, the users are presented with the information - the recommended question with a score. This score is not indicative of the lexical similarity of the recommended question and the asked question. Instead, it is a measure of how many times and at which level is a match found. It is a cumulative score of the number of representational matches of the recommended questions made in the knowledge base. For scoring the levels we made an assumption that each levels is in descending order; A match in Level 1 is more important and highly weighted than a match in Level 2 and so on for other levels. Table 4.4 indicates the scores assigned at each level. This allocation of score is to represent a dummy confidence score for the recommended questions, it is also allocated to distinguish between two or more recommended question and how probable is that a recommended question is semantically similar to a asked question based on the matches found at the subsequent levels.

Levels	Score Assigned
Level 1	100
Level 2	90
Level 3	70
Level 4	50
Level 5	10

Table 4.4: Scores assigned to each Level

Figure 4.15 demonstrates the levels of matches being performed and on processing through all the levels of SPARQL query execution for each triple, the presentation of the recommended question along with the cumulative score which is obtained by the matches made at each level. It is to be noted that when a user question is encountered by the system, the OpenIE approach of triple generation is used to generate the triples.



```
Enter your question:
Does anyone know about the best squash player in the globe?
Subject: anyone Predicate: do_know_about       Object:         best_squash_player_in_globe
No questions at Level....1
No questions at Level....2
No questions at Level....3
No questions at Level....4
No questions at Level....5
Subject: anyone Predicate: do_know_about       Object:         squash_player
No questions at Level....1
No questions at Level....2
No questions at Level....3
No questions at Level....4
No questions at Level....5
Subject: anyone Predicate: do_know_about       Object:         best_squash_player
No questions at Level....1
No questions at Level....2
No questions at Level....3
No questions at Level....4
No questions at Level....5
Subject: best_squash_player    Predicate: in   Object:         globe
No questions at Level....1
Subject: anyone Predicate: do_know_about       Object:         squash_player_in_globe
No questions at Level....1
No questions at Level....2
No questions at Level....3
No questions at Level....4
No questions at Level....5
question: Who is the best squash player in the world
score: 90

Do you find this question similar? (y/n)
y
Updating Ontology with new knowledge!
```

Figure 4.15: Demonstration of triple generation, search and presentation of the recommended question to the user with a score.

4.8 Summary

The extraction of semantic triples, modelling them using RDF and thereby using these triples to construct queries for expressing information need over RDF data was accomplished. Unstructured data was converted to structured data and used to store and query machine understandable metadeta. User knowledge was contained in the knowledge base with the provision of enriching the triple store. The implementation of the proposed approach suggests that it is possible to identify semantically similar natural language sentences using linked data to a considerable extent, however, the application at present is in it's nascent stages and there are room for improvements. Analysis of the triples generated and ways of generating triples of better quality is a present limitation and better engineering decisions related to this module could enrich not only the quality of the triples generated but also the knowledge contained in the knowledge base.

The evaluation and the findings for this proof-of-concept approach has been presented in the forthcoming chapter.

Chapter 5

Evaluation

For evaluating this prototype formative user-centered evaluation has been conducted. This chapter describes how the evaluation was conducted, what were the findings, which metrics were used to evaluate the system and what was the outcome of these evaluations. The chapter concludes with a critical analysis of the evaluation process and threats to validation in the process.

5.1 User Evaluation

Since the evaluation of semantic web technologies based approach of identifying semantically similar question is not a straightforward task as there are no available gold-standard datasets for comparison, human intervention is required and to the best of our knowledge there are no standard criteria which is defined to carry out manual evaluation.

In our case judgement is primarily based on the ratio between the semantic similarity of the recommended question versus the asked question. In the evaluation process, the person who framed a question made the judgment as well.

5.2 Metrics

An overall agreement was required as there were a number of human participants. The number of human judges and the categories (whether a recommended question is semantically similar; yes or no) are considered.

The judgement of the semantic accuracy with respect to categories (Yes/No based on Semantic Similarity) by the user who asked this question (participant's judgement) has been noted.

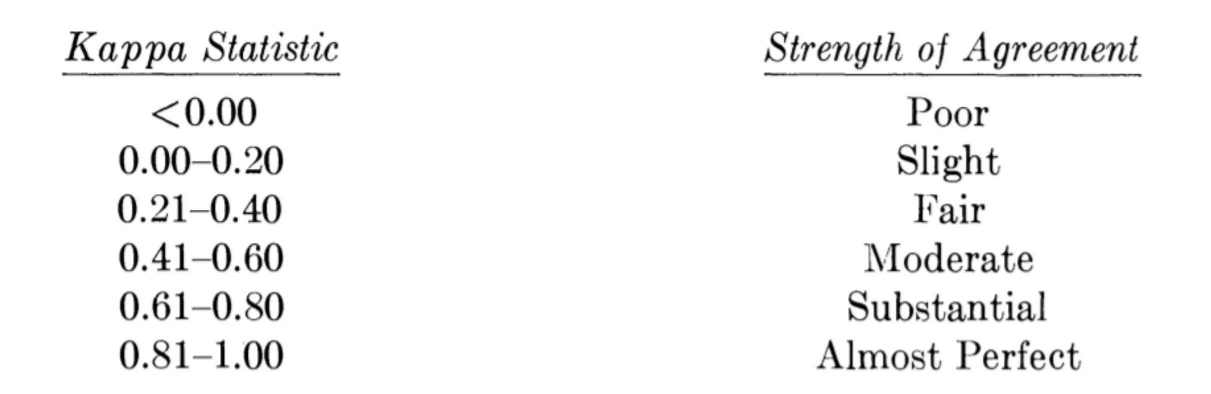
Kappa-based metric [68] for inter-rater agreement is used. There are possibilities of bias and chance among the agreement between evaluation participants,therefore, to avoid this, we use the

Kappa [68] value to determine the inter-rater agreement between the observed data and the prior data. To obtain the value of Kappa the following formula is used.

$$K_{free} = \frac{\left[\frac{1}{Nn(n-1)}\left(\sum_{i=1}^{N}\sum_{j=1}^{k} n_{\mathrm{ij}^2} - Nn\right)\right] - \left[\frac{1}{k}\right]}{1 - \left[\frac{1}{k}\right]} \tag{5.1}$$

In this equation, k is the number of categories, N is the number of cases, n is the number of evaluation participants (rater), and n_{ij} is an array of ratings where n_ij[i][j] evaluators assigned case i to category j.

In the following Figure 5.1 the Kappa statistics and the division of the strength of agreement is represented.



Kappa Statistic	*Strength of Agreement*
<0.00	Poor
0.00–0.20	Slight
0.21–0.40	Fair
0.41–0.60	Moderate
0.61–0.80	Substantial
0.81–1.00	Almost Perfect

Figure 5.1: Useful benchmarks related to Kappa Statistics [8]

We also adopt the measures used in a study by Zhu et al. [1] to evaluate the performance of our proposed approach represented in Table 5.2.

Performance Measures		
Category	Measure	Description
Relevance Based	Success Rate	A binary measure (i.e., success or not) represents completion of each task.
Interaction Based	Search Time	Time taken to complete a search task
	Query size	The number of issued queries to complete a search task.

Table 5.1: Measures used in evaluation [1]

We measure the success rate by the ability of the application to generate semantic triples from the given text (user posed questions). The search time is the time taken for the application to recommend a question and the query size is the number of queries required to find a semantically similar match.

Finally we present a ratio between the following

- True Positive(TP)- Questions which are semantically similar and recommended by the system during the evaluation process.
- False Positive(FP) - Questions which are semantically not similar but still recommended by the system during evaluation process.

Since we have the values for TP and FP we can calculate the Precision of the system by the following equation.

$$Precision = \frac{TP}{TP+FP} \tag{5.2}$$

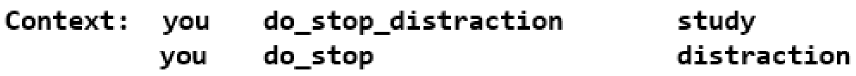
We have considered the above-mentioned metrics to evaluate our system.

5.2.1 Subjects

Fifteen participants were recruited using a convenience-sampling method. The participants were sent a recruitment message for volunteering . To adjust the research setting a pilot-test was conducted. During the test, through interviews with potential participants we found that most of the participants were users of community question answering sites, who have either posted a question or posted an answer or have posted questions as well as answered questions and those who used such sites to get a solution for their query.

5.2.2 Tasks

The tasks designed were as such wherein users were provided with partial information, where they were shown a set of contexts based on which they were asked to frame questions. For example, they were shown the below set of semantic triples as shown in Fig5.2 and asked to frame questions, the suggestion that there is no right or wrong answers were indicated before the test commenced. To reduce the variance of task complexity, each of the task sets were curated to be as similar as possible. Each task set had 4 contexts based on which the participants were asked to frame questions.



```
Context:  you   do_stop_distraction    study
          you   do_stop                distraction
```

Figure 5.2: Example of context presented to participants for framing questions.

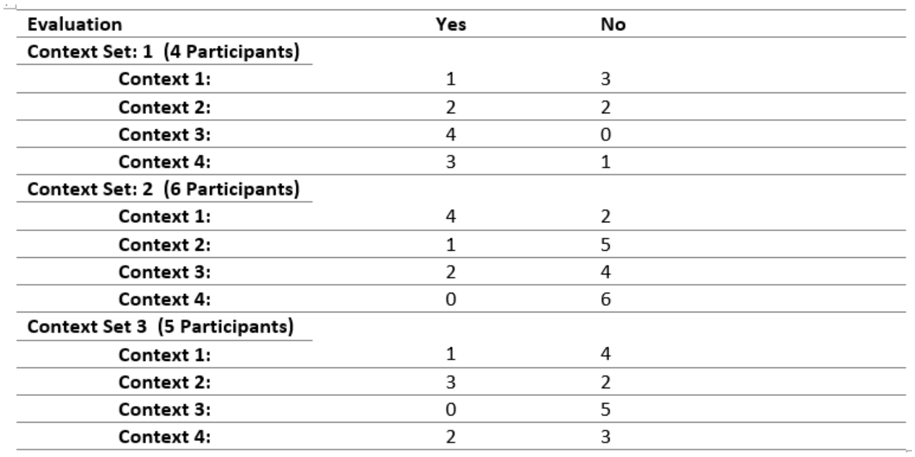
There were three set of contexts with each context set having four contexts. The following are the number of human participants who participated in each context set.

Context Set	Number of Participant
Set 1	4
Set 2	6
Set 3	5

Table 5.2: Number of participants in each context set

5.2.3 Findings

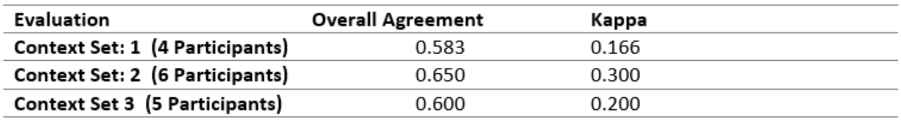
We present the result of the Kappa evaluation below. Figure 5.3 indicates the rating of the users based on each context of a context set.



Evaluation	Yes	No
Context Set: 1 (4 Participants)		
Context 1:	1	3
Context 2:	2	2
Context 3:	4	0
Context 4:	3	1
Context Set: 2 (6 Participants)		
Context 1:	4	2
Context 2:	1	5
Context 3:	2	4
Context 4:	0	6
Context Set 3 (5 Participants)		
Context 1:	1	4
Context 2:	3	2
Context 3:	0	5
Context 4:	2	3

Figure 5.3: Rating of the users based on the presented results in two categories; Yes indicates semantically similar result, No indicates results which are not semantically similar as decided by the human participants.

The overall agreement and the Kappa value is indicated in Figure 5.4. The average overall agreement obtained is 0.61 which indicates that the human participants for evaluation shares decisions quite often. The average Kappa value obtained is 0.22 which seems to be 'Fair' based on the benchmark outlined in Figure 5.1. These results indicate that the data presented to the users were coherent and it also signifies that the evaluation was not influenced by chance.



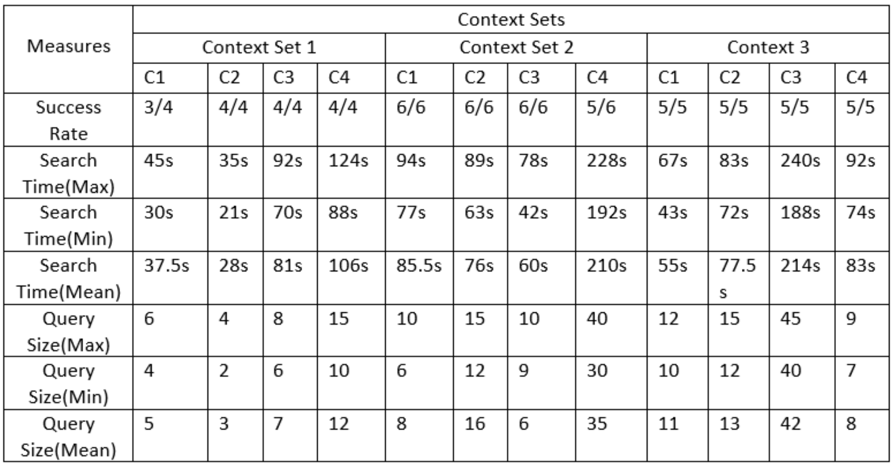
Evaluation	Overall Agreement	Kappa
Context Set: 1 (4 Participants)	0.583	0.166
Context Set: 2 (6 Participants)	0.650	0.300
Context Set 3 (5 Participants)	0.600	0.200

Figure 5.4: The overall agreement score and the Kappa value for each of the context set.

The evaluation for the success rate, search time and query size was measured using a system with a CPU 2.20GHz and 8GB RAM.

We calculate the success rate of each event during evaluation and measure the search time and the number of queries (Query size) required to present a result to the users. We find that in most cases the system was able to generate a result for the users except in cases when the questions formed by the users was not in the form of Noun-Verb-Noun tags representation which is required for the generation of the triples. Since, no triples were generated for such questions the system could not proceed and was not able to present data to the users. The mean search time, if all contexts are to be taken into account lies between 37.5 seconds to 214 seconds, which indicates that search time is dependent on the context and the nature of questions being formed by the users. If the word length of the question being formed is large and the semantic triples generated due to the large length of the questions is much more than for small sentences, the search time increases significantly.

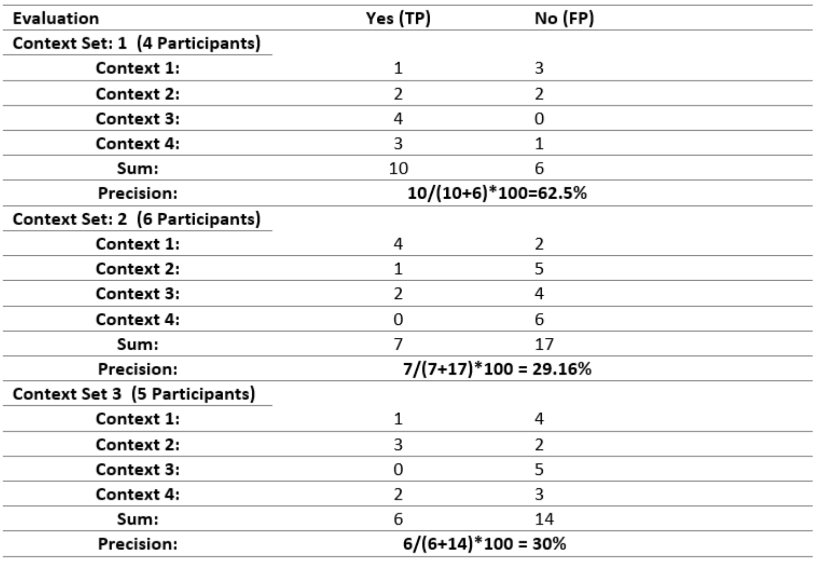
There are five levels of search in our approach and the query size is representative of how many levels of validations were required to present the users with a result. For example, if the system is able to present the users with a result after search in Level 1; the query size is 1. If the system is able to present the users with a result after search in Level 5, the query size is 5. Even after, Level 5, if the system doesn't present the users with a result and moves on to the next set of triples and starts the search at Level 1; the query size in this case is 6(5 queries till Level 5 of the previous search attempt and 1 query for Level 1 search for the current set of triples). Fig 5.5 represent the quantitative results of the success rate, search and query time for each context.



Measures	Context Sets											
	Context Set 1				Context Set 2				Context 3			
	C1	C2	C3	C4	C1	C2	C3	C4	C1	C2	C3	C4
Success Rate	3/4	4/4	4/4	4/4	6/6	6/6	6/6	5/6	5/5	5/5	5/5	5/5
Search Time(Max)	45s	35s	92s	124s	94s	89s	78s	228s	67s	83s	240s	92s
Search Time(Min)	30s	21s	70s	88s	77s	63s	42s	192s	43s	72s	188s	74s
Search Time(Mean)	37.5s	28s	81s	106s	85.5s	76s	60s	210s	55s	77.5s	214s	83s
Query Size(Max)	6	4	8	15	10	15	10	40	12	15	45	9
Query Size(Min)	4	2	6	10	6	12	9	30	10	12	40	7
Query Size(Mean)	5	3	7	12	8	16	6	35	11	13	42	8

Figure 5.5: Success rate, Search Time and the Number of Queries used for each context .

Finally, the precision of the system has been calculated based on the semantic similarity judgment by the users. The 'Yes' response from the users on whether the recommended question by the system is semantically similar is considered as the 'True Positive' values; a question which is semantically similar with respect to the asked question and presented to the user. 'No' response from the users are considered as the 'False Positive' value ; a question which is not semantically similar yet recommended to the users. Figure 5.6 shows the detailed calculation for the precision of the system based on each context.



Evaluation	Yes (TP)	No (FP)
Context Set: 1 (4 Participants)		
Context 1:	1	3
Context 2:	2	2
Context 3:	4	0
Context 4:	3	1
Sum:	10	6
Precision:	**10/(10+6)*100=62.5%**	
Context Set: 2 (6 Participants)		
Context 1:	4	2
Context 2:	1	5
Context 3:	2	4
Context 4:	0	6
Sum:	7	17
Precision:	**7/(7+17)*100 = 29.16%**	
Context Set 3 (5 Participants)		
Context 1:	1	4
Context 2:	3	2
Context 3:	0	5
Context 4:	2	3
Sum:	6	14
Precision:	**6/(6+14)*100 = 30%**	

Figure 5.6: Precision of the system based on each context set.

The kind of context, the nature of user formed questions, the semantic triples generated from the user formed questions and the knowledge in the triple store are all contributing factors in determining precision of the system. Context set 1 has more precision as the nature of the user formed question generated similar semantic triples most of the times which were already present in the knowledge base. Just by having more number of similar questions in the knowledge base is not a clear indicator of good precision in our case, existence of similar semantic triples tagged to the appropriate questions and such triples getting generated during the processing of natural language questions encourages good precision. Our observations from the study suggests that if the user formed question is as such that the NLP tools employed to generate semantic triples are similar to the already stored semantic triples in the knowledge base, the precision of the recommended results increases significantly.

Observations from our evaluation indicates that our approach is human understandable. Given a choice of more words in a context for forming questions, the variance of the questions formed by the user increased. It was observed that the participants spent more time in forming questions when presented with more options (words in a context). Few participants formed multiple questions from the same context easily when presented with more options. Restricting the words in a context gave less freedom to the participants to form questions and the questions they formed usually generated the semantic triples which were present in the knowledge base.

5.3 Critical Analysis

It is expensive and unfeasible in terms of time and effort required by human participants to evaluate a large number of instances. Moreover, a fixed domain facilitates the recruitment of better human participants, since, in our proposed approach we have made an open domain attempt, pre-defined domain specific rules as well as evaluation by human-experts in specific domains is absent. Strengths and weaknesses of the system from several perspectives were examined and the experimental design and the results represented in this study can serve as a useful guideline for further studies on the same topic of identifying semantically similar question. However, some aspects which could be influencing the agreement or disagreement among participants are as follows:

- The abstract context presented to the participants and based on the understanding of the context or the core idea, variance in the formulation of the question by the participants and thereby influencing the judgement whether such a question is semantically similar against the recommended questions.
- The reliance on Natural language processing (NLP) tools for relationship extraction and the interpretation of the results from such tools by the human participants was an influencing factor.

For example, a series of semantic triples generated from a given question incited confusion and influenced their decision of marking a specified case as negative.

- Traditional knowledge of Information Retrieval concepts such as n-grams and stop words influenced the decision of the judgement. For example, prepositions in our approach is used as a predicate to establish a relationship between a subject and an object, however, it is considered as good candidates for stop words in Information Retrieval, these unfamiliar understanding by the participants impacted the evaluation.

Even though the count of participants were less and there were issues in judgment by the participants our proposed approach to identify semantically similar questions by using NLP tools and Linked Data principles holds significant promise.

5.4 Summary

The outcome of the evaluation indicates that identifying semantically similar unstructured text with the help of linked data approach is a possibility. If both domain-specific or domain independent data modelling is done based on the given problem statement, this approach is extensible and maintainable. There are various limitations of this approach, however, the primary user centered evaluation confirms that this approach is human understandable and with the help of knowledge experts could be fine-tuned in the event of semantic drift within the knowledge base making the system extendable and relevant. We believe based on the pilot methodology and the preliminary results obtained from our evaluation, the proposed approach is viable. This approach could with multiple layers of analysis-each layer focusing on enriching the knowledge base and recommending semantically similar questions holds promise. It facilitates not only identifying semantically similar questions from unstructured text but also in creation of a reusable, extendable and human-understandable knowledge base which could be reasoned upon and used for gaining insights.

Chapter 6

Conclusion

The study conducted to answer the research question as mentioned in section 1.3 yeilded interesting and insightful findings. In this chapter we discuss the possibilities and limitations of this apporach, recommendations for future work and finally we summarize stating how successful was this dissertation.

6.1 Possibilities

In the domain of question answering and with respect to the specific case of our work of identifying similar questions, linked data approach has a great potential and has shown significant promise during the course of our research. It is the opinion of the author that with additional refinements, investigation and experimentation of this approach a web-scale application can be realized which would aid in dynamic creation and enrichment of knowledge base which could be exploited to gain reasonable insights for general and specific purposes. This approach has sufficient potential in the area of knowledge harvesting using a structured approach. The growing advancement in the area of NLP techniques for relationship extraction and the infusion of results from such techniques to model data in the format as specified by W3C standards thereby using such data to gain information cannot be ignored or undermined. This approach also facilitates the avenue of data monitoring which allows in maintaining consistent standard of data quality.

6.2 Limitations

We would like to discuss on two types of limitations, primarily the engineering limitations which can be fine-tuned with more time and effort and the specific limitations of this approach. In our current approach the generation of triples and the corresponding synonyms generated from those triples is highly dependent on the NLP tools and the WordNet API that has been used to gather data for construction of the knowledge base. Research related to the quality of triples generation and comparison between various state-of-the art techniques to choose the best technique or customize a current technique would help in increasing the quality of the generated triples. Our current approach does not focus on the ranking of triples based on the semantic similarity of the sentence from which such triples are generated. This is an engineering limitation and can be solved with state-of-the art algorithms for classification and clustering. Another important feature of spelling correction before using generated triples to model a knowledge base so as to decrease the amount of noise in the knowledge base could be introduced as such a feature doesn't exist within the scope of our work. A more thorough analysis of WordNet and to retrieve only meaningful synonyms and the related meronym, hyponym and thereby building a customized synonym engine would facilitate in creating robust building blocks for the knowledge base. Monitoring, maintaining and updating the triple store is required by the knowledge experts from time to time.

6.3 Future Work

Limitations of the present approach and the possible solution has been discussed in the previous section. In this section we would like to provide few recommendations for future work. There are various options by which the present work could be expanded based on the current findings. The design of the knowledge base from a technical perspective could be further developed as a five star linked data application which would facilitate inter-linking of data as well as providing richer context for the users. Such design developments, would encourage in using more external vocabularies as well as publishing data on the Linked Open Data cloud. Given that the findings of the current study emphasizes on the different ways in which users forms questions depending on what context is shown to them. It would be interesting to investigate as well as to compare the contexts shown to the users (and the user posed questions based on these contexts) to derive richer semantics which could then be thoroughly analyzed to be included in the knowledge base. For future research, criteria for selection of participants having knowledge in the fields of cognitive sciences, human reasoning or people involved in the study of human behaviour and mental processes could be recruited as they would be able to provide richer variety of contexts. Inclusion of such participants would greatly help further research on this proposed structured approach of identifying semantically similar questions using linked data.

6.4 Summary

An end-to-end system for the identification of semantically similar natural language questions has been developed during the course of this research. The results of this approach were user evaluated and the findings demonstrates that such an approach is human as well as machine understandable. This approach also indicates that it is more dependent on the human-understandable parameters rather than just machine-understandable parameters.

Despite the fact that the findings of this research were limited to identifying semantically similar questions rather than question answering, this work has witnessed a development of a novel application for using linked data approach in finding semantically similar natural language sentences. The fact that data from CQA sites could be processed using NLP technique, modelled in a structured format such as RDF and used to query information using SPARQL is a significant accomplishment of this work. It also provides a meaningful contribution in the area of community question answering using linked data.

Bibliography

[1] Y. Zhu, M. C. Kim, and E. Yan, "Evaluating interactive bibliographic information retrieval systems: A user-centered approach," *Proceedings of the Association for Information Science and Technology*, vol. 55, no. 1, pp. 628–637, 2018.

[2] C. Shah, S. Oh, and J. S. Oh, "Research agenda for social qa," *Library Information Science Research*, vol. 31, no. 4, pp. 205 – 209, 2009.

[3] K. Bae and Y. Ko, "Improving question retrieval in community question answering service using dependency relations and question classification," *Journal of the Association for Information Science and Technology*, 2019.

[4] G. Wang, K. Gill, M. Mohanlal, H. Zheng, and B. Y. Zhao, "Wisdom in the social crowd: an analysis of quora," in *WWW*, 2013.

[5] M. R. Martin and S. Auer, "Categorisation of semantic web applications," 2010.

[6] T. Wei, Y. Lu, H. Chang, Q. Zhou, and X. Bao, "A semantic approach for text clustering using wordnet and lexical chains," *Expert Systems with Applications*, vol. 42, no. 4, pp. 2264 – 2275, 2015.

[7] R. Glauber and D. B. Claro, "A systematic mapping study on open information extraction," *Expert Syst. Appl.*, vol. 112, pp. 372–387, 2018.

[8] J. R. Landis and G. G. Koch, "The measurement of observer agreement for categorical data," *Biometrics*, vol. 33, pp. 159–174, 1977.

[9] L. A. Adamic, J. Zhang, E. Bakshy, and M. S. Ackerman, "Knowledge sharing and yahoo answers: Everyone knows something," in *Proceedings of the 17th International Conference on World Wide Web*, WWW '08, (New York, NY, USA), pp. 665–674, ACM, 2008.

[10] A. Baltadzhieva and G. Chrupa, "Question quality in community question answering forums: A survey," *SIGKDD Explor. Newsl.*, vol. 17, pp. 8–13, Sept. 2015.

[11] J. Jeon, W. B. Croft, J. H. Lee, and S. Park, "A framework to predict the quality of answers with non-textual features," in *SIGIR*, 2006.

[12] X. Xue, J. Jeon, and W. B. Croft, "Retrieval models for question and answer archives," in *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '08, (New York, NY, USA), pp. 475–482, ACM, 2008.

[13] E. Agichtein, C. Castillo, D. Donato, A. Gionis, and G. Mishne, "Finding high-quality content in social media," in *Proceedings of the 2008 International Conference on Web Search and Data Mining*, WSDM '08, (New York, NY, USA), pp. 183–194, ACM, 2008.

[14] A. Shtok, G. Dror, Y. Maarek, and I. Szpektor, "Learning from the past: Answering new questions with past answers," in *Proceedings of the 21st International Conference on World Wide Web*, WWW '12, (New York, NY, USA), pp. 759–768, ACM, 2012.

[15] B. Li and I. King, "Routing questions to appropriate answerers in community question answering services," in *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, (New York, NY, USA), pp. 1585–1588, ACM, 2010.

[16] L. Chen, "Understanding and exploiting user intent in community question answering," in *PhD thesis, Birkbeck, University of London*, 2014.

[17] M. Asaduzzaman, A. S. Mashiyat, C. K. Roy, and K. A. Schneider, "Answering questions about unanswered questions of stack overflow," in *2013 10th Working Conference on Mining Software Repositories (MSR)*, pp. 97–100, May 2013.

[18] P. Jurczyk and E. Agichtein, "Discovering authorities in question answer communities by using link analysis," in *Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management*, CIKM '07, (New York, NY, USA), pp. 919–922, ACM, 2007.

[19] M. Qu, G. Qiu, X. He, C. Zhang, H. Wu, J. Bu, and C. Chen, "Probabilistic question recommendation for question answering communities," in *Proceedings of the 18th International Conference on World Wide Web*, WWW '09, (New York, NY, USA), pp. 1229–1230, ACM, 2009.

[20] D. Horowitz and S. D. Kamvar, "The anatomy of a large-scale social search engine," in *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, (New York, NY, USA), pp. 431–440, ACM, 2010.

[21] E. Mendes Rodrigues and N. Milic-Frayling, "Socializing or knowledge sharing?: Characterizing social intent in community question answering," in *Proceedings of the 18th ACM Conference*

on Information and Knowledge Management, CIKM '09, (New York, NY, USA), pp. 1127–1136, ACM, 2009.

[22] T. Berners-Lee, J. Hendler, and O. Lassila, "The semantic web," *Scientific American*, pp. 34–43, 2001.

[23] D. A. Ferrucci, "Introduction to "this is watson"," *IBM Journal of Research and Development*, vol. 56, pp. 1:1–1:15, May 2012.

[24] J.-P. Ng and M.-Y. Kan, "Qanus: An open-source question-answering platform," *ArXiv*, vol. abs/1501.00311, 2015.

[25] V. Lopez, M. Pasin, and E. Motta, "Aqualog: An ontology-portable question answering system for the semantic web," in *The Semantic Web: Research and Applications* (A. Gómez-Pérez and J. Euzenat, eds.), (Berlin, Heidelberg), pp. 546–562, Springer Berlin Heidelberg, 2005.

[26] Soojong Lim, Euisuk Jung, and Myoung-Gil Jang, "Dependency relation analysis using case frame for encyclopedia question-answering system," in *30th Annual Conference of IEEE Industrial Electronics Society, 2004. IECON 2004*, vol. 3, pp. 3120–3124 Vol. 3, Nov 2004.

[27] Robin D. Burke, Kristian J. Hammond, Vladimir Kulyukin, Steven L. Lytinen, Noriko Tomuro, and Scott Schoenberg, "Question answering from frequently asked question files: Experiences with the faq finder system," in *AI Magazine, Vol 18 No 2: Summer 1997*, AI Magazine, 1997.

[28] C. Fellbaum, "Wordnet: An electronic lexical database.," in *Cambridge, MA*, MIT Press, 1998.

[29] Y.-S. Lai, K.-A. Fung, and C.-H. Wu, "Faq mining via list detection," in *Proceedings of the 2002 Conference on Multilingual Summarization and Question Answering - Volume 19*, MultiSumQA '02, (Stroudsburg, PA, USA), pp. 1–7, Association for Computational Linguistics, 2002.

[30] A. J. Gill, L.-C. Umr, and R. M. French, "Level of representation and semantic distance: Rating author personality from texts," 2006.

[31] K. Zhang, W. Wu, H. Wu, Z. Li, and M. Zhou, "Question retrieval with high quality answers in community question answering," in *CIKM*, 2014.

[32] Z. Ji, F. Xu, B. Wang, and B. He, "Question-answer topic model for question retrieval in community question answering," in *CIKM*, 2012.

[33] J. D. Lafferty and C. Zhai, "Document language models, query models, and risk minimization for information retrieval," in *SIGIR*, 2001.

[34] J.-R. Wen, J.-Y. Nie, and H. Zhang, "Query clustering using user logs," *ACM Trans. Inf. Syst.*, vol. 20, pp. 59–81, 2002.

[35] A. Tombros, R. Villa, and C. J. van Rijsbergen, "The effectiveness of query-specific hierarchic clustering in information retrieval," *Inf. Process. Manage.*, vol. 38, pp. 559–582, 2002.

[36] G. Zhou, L. Cai, J. Zhao, and K. Liu, "Phrase-based translation model for question retrieval in community question answer archives," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, (Stroudsburg, PA, USA), pp. 653–662, Association for Computational Linguistics, 2011.

[37] K. Liu, L. Xu, and J. Zhao, "Opinion target extraction using word-based translation model," in *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, (Stroudsburg, PA, USA), pp. 1346–1356, Association for Computational Linguistics, 2012.

[38] J. Gao, X. He, and J.-Y. Nie, "Clickthrough-based translation models for web search: From word models to phrase models," in *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, (New York, NY, USA), pp. 1139–1148, ACM, 2010.

[39] W. Zhang, Y. Li, and S. Wang, "Learning document representation via topic-enhanced lstm model," *Knowl.-Based Syst.*, vol. 174, pp. 194–204, 2019.

[40] M. Yang, W. Tu, Q. Qu, W. Zhou, Q. Liu, and J. Zhu, "Advanced community question answering by leveraging external knowledge and multi-task learning," *Knowl.-Based Syst.*, vol. 171, pp. 106–119, 2019.

[41] Y. Liu, J. Bian, and E. Agichtein, "Predicting information seeker satisfaction in community question answering," in *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '08, (New York, NY, USA), pp. 483–490, ACM, 2008.

[42] A. Mockus and J. D. Herbsleb, "Expertise browser: A quantitative approach to identifying expertise," in *In 2002 International Conference on Software Engineering*, pp. 503–512, ACM Press, 2002.

[43] J. Guo, S. Xu, S. Bao, and Y. Yu, "Tapping on the potential of q&
a community by recommending answer providers," in *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, CIKM '08, (New York, NY, USA), pp. 921–930, ACM, 2008.

[44] M. Liu, Y. Liu, and Q. Yang, “Predicting best answerers for new questions in community question answering,” in *Proceedings of the 11th International Conference on Web-age Information Management*, WAIM’10, (Berlin, Heidelberg), pp. 127–138, Springer-Verlag, 2010.

[45] G. Zhou, L. Qian, and J. Fan, “Tree kernel-based semantic relation extraction with rich syntactic and semantic information,” *Inf. Sci.*, vol. 180, pp. 1313–1325, 2010.

[46] O. Etzioni, M. Banko, S. Soderland, and D. S. Weld, “Open information extraction from the web,” *Commun. ACM*, vol. 51, pp. 68–74, 2008.

[47] S. Brin, “Extracting patterns and relations from the world wide web,” in *WebDB*, 1998.

[48] D. E. Rose and D. Levinson, “Understanding user goals in web search,” in *WWW*, 2004.

[49] A. Broder, “A taxonomy of web search,” *SIGIR Forum*, vol. 36, pp. 3–10, Sept. 2002.

[50] D. Moldovan, S. Harabagiu, M. Pasca, R. Mihalcea, R. Girju, R. Goodrum, and V. Rus, “The structure and performance of an open-domain question answering system,” in *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, ACL ’00, (Stroudsburg, PA, USA), pp. 563–570, Association for Computational Linguistics, 2000.

[51] E. Hovy, L. Gerber, U. Hermjakob, C.-Y. Lin, and D. Ravichandran, “Toward semantics-based answer pinpointing,” in *Proceedings of the First International Conference on Human Language Technology Research*, HLT ’01, (Stroudsburg, PA, USA), pp. 1–7, Association for Computational Linguistics, 2001.

[52] S.-Y. Yang, C.-L. Hsu, D. liang Lee, and L. Y. Deng, “Faq-master: an ontological multi-agent system for web faq services,” 2008.

[53] Y. Liu, S. Li, Y. Cao, C.-Y. Lin, D. Han, and Y. Yu, “Understanding and summarizing answers in community-based question answering services,” in *Proceedings of the 22Nd International Conference on Computational Linguistics - Volume 1*, COLING ’08, (Stroudsburg, PA, USA), pp. 497–504, Association for Computational Linguistics, 2008.

[54] N. Dandekar, “Semantic question matching with deep learning,” 2017.

[55] P. G. E. Hein, “Constructivist learning theory,” *CECA (International Committee of Museum Educators) Conference*, 1991.

[56] D. Soni, R. L. Nord, and C. Hofmeister, “Software architecture in industrial applications,” *1995 17th International Conference on Software Engineering*, pp. 196–196, 1995.

[57] D. Rusu, L. Dali, B. Fortuna, M. Grobelnik, and D. Mladení, "Triplet extraction from sentences," 2007.

[58] G. Angeli, M. J. J. Premkumar, and C. D. Manning, "Leveraging linguistic structure for open domain information extraction," in *ACL*, 2015.

[59] L. Meng, R. Huang, and J. Gu, "An effective algorithm for semantic similarity metric of word pairs," *International Journal of Multimedia and Ubiquitous Engineering*, vol. 8, no. 2, pp. 1–12, 2013.

[60] S. Iyer, N. Dandekar, and K. Csernai, "First quora dataset release: Question pairs," in *Quora*, 2017.

[61] M. Fossati, E. Dorigatti, and C. Giuliano, "N-ary relation extraction for simultaneous t-box and a-box knowledge base augmentation," *Semantic Web*, vol. 9, pp. 413–439, 2017.

[62] N. Jaywant, S. Shetty, and V. Musale, "Digital identity based recommendation system using social media," in *2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)*, pp. 288–292, July 2016.

[63] S. Riedel, L. Yao, A. McCallum, and B. M. Marlin, "Relation extraction with matrix factorization and universal schemas," in *HLT-NAACL*, 2013.

[64] M. J. Hosseini, N. Chambers, S. Reddy, X. R. Holt, S. B. Cohen, M. Johnson, and M. Steedman, "Learning typed entailment graphs with global soft constraints," *Transactions of the Association for Computational Linguistics*, vol. 6, pp. 703–717, 2018.

[65] L. Romano, M. Kouylekov, I. Szpektor, I. Dagan, and A. Lavelli, "Investigating a generic paraphrase-based approach for relation extraction," in *EACL*, 2006.

[66] A. Carlson, J. Betteridge, B. Kisiel, B. Settles, E. R. Hruschka, and T. M. Mitchell, "Toward an architecture for never-ending language learning," in *AAAI*, 2010.

[67] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky, "The Stanford CoreNLP natural language processing toolkit," in *Association for Computational Linguistics (ACL) System Demonstrations*, pp. 55–60, 2014.

[68] J. J. Randolph, "Free-marginal multirater kappa (multirater k[free]): An alternative to fleiss' fixed-marginal multirater kappa.," 2005.

Appendix

...