Using Dynamic Knowledge Graph for Fake News Early Detection

Albertus Andito

Candidate No. 198910

Project Supervisor: Dr. Julie Weeds

BSc Computer Science (with an industrial placement year)
Department of Informatics
University of Sussex

May 2021

Statement of Originality

This report is submitted as part requirement for the degree of BSc Computer Science (with an industrial placement year) at the University of Sussex. It is the product of my own labour except where indicated in the text. The report may be freely copied and distributed provided the source is acknowledged. I hereby give permission for a copy of this report to be loaned out to students in future years.

Signature:

Albertus Andito

Acknowledgements

I would like to firstly thank my supervisor, Dr. Julie Weeds, who has been supporting me throughout this final year project. The knowledge and wisdom that she has given are invaluable to the completion of this project. Her enthusiasm has kept me inspired.

I would also like to express my gratitude to my family, who is always there for me and has always motivated me even from far away. I also thank my friends for bringing joy and support.

Summary

With the rapid rate of misinformation dissemination that is happening right now, it is important to be able to detect fake news as early as possible. In order to do that, the collection of facts that are used as the ground truth needs to be updated all the time. By having the latest facts in hand, a more accurate fact-checking can be performed, which will verify if a news is true or fake. In this project, the collection of facts is represented by a knowledge graph.

This project aims to develop a fake news detection system that uses a dynamic knowledge graph, which stores the ground truth, to help identify fake news. The system is able to extract facts, in the form of semantic triples, from news articles and update the knowledge graph accordingly with the facts. The system also has fact-checking algorithms that can infer if matches of a triple can be found in the knowledge graph or not. As the focus of the system is to assist human verifiers in doing their jobs, this system can be accessed through a web-based user interface.

An evaluation of the system shows that the quality of the triples hugely affects the performance of the fact-checker. The evaluation was done by feeding the knowledge graph with real news articles and fact-checking other real news of the same topic from other sources as a form of verification. Although the evaluation proves that there is a lot of room for improvements in terms of fact-checking, as a framework, the system has introduced the steps in the pipeline and has managed to do them well as has been also shown through usability testing.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Identifying fake news and performing fact-checking are not trivial or straightforward to do, especially when one wants to do them at an early stage, before the fake news propagates widely. This is because news articles are often published at a rapid rate. Therefore, the collection of facts, that are used as the ground truth in the fact-checking process, needs to be updated accordingly all the time. In addition to that, there is the timeliness aspect of news, which can result in having a ground truth that is now obsolete because it has been superseded by the information in a latest news.

This project overcomes those problems by presenting a system that dynamically expands the ground truth collection, in the form of knowledge graph, with facts extracted from trusted news articles. The fact-checking can then be performed by querying the so-called dynamic knowledge graph to get some inference of truthfulness.

1.1 Motivation

In the recent years, the dissemination of fake news, particularly in social media, has become a serious problem that is affecting our society. Most notably, it was estimated in [1] that the average American saw at least one fake news related to the 2016 United States presidential election in the months preceding the election. The study confirmed that most of the widely shared fake news were in favour of Donald Trump, which might have impacted the election results.

More recently, when the world was taken aback by the emergence of COVID-19, a huge amount of misinformation around the disease, in addition to the virus responsible for the disease itself, was spread widely and rapidly. During the pandemic, it has caused people to overreact, such as by hoarding goods, as well as to underreact, for example by engaging in risky activities which resulted in unintentional spread of the virus [2]. The sudden influx of new information (and misinformation) makes it difficult for many people to quickly differentiate false facts from true facts.

To identify fake news and deal with their wide propagation, many approaches have been developed. Some of them are manual fact-checking methods, which are done by organisations such as Snopes[1] and Fullfact[2]. Other approaches are categorised as automatic fake news detectors. For example, methods that use classical machine learning approaches such as the one experimented in [3] and methods that use knowledge graph embeddings, which are introduced in [4]. They generally work well for detecting misinformation on news that were published in the past.

There are several categories of automatic fake news detection methods that are usually used. Some methods rely on the way fake news are being propagated on social media. For example, in [5], users' posting behaviour is used as a feature to assess the credibility of an information on Twitter. Similarly, in [6], the features come from user engagements and the relationships between users. However, those methods are not able to detect fake news at an early stage because they need to have social context information, which might not yet available at that time [7]. There are also methods which detect fake news by looking at its writing style, such as the method in [7]. Unfortunately, as discussed by [8], they cannot be applied for all news topics and they might become less accurate in the future because deceptive writing styles are always evolving.

Thus, methods that are potentially effective to detect fake news at an early stage are the ones that primarily rely only on the news content. This project explored an instance of such methods, which is the knowledge-based method. In this method, the news content are represented as a set of SPO (Subject, Predicate, Object) triples that are stored in a knowledge graph. An example of an SPO triple is `(John_Doe, ignore, Social_distancing)`, which is extracted from the sentence "John Doe ignored social distancing." Knowledge graph is defined as a graph structure that uses SPO triples as facts representation, where entities (Subjects and Objects) and relationships (Predicates) are represented as nodes and edges in the graph, respectively [9]. Having the knowledge graph as the collection of ground truth, the fake news detection is done by comparing the information from to-be-verified article with the knowledge graph using various fact-checking techniques, which was done in [10] and [11], for example.

However, using a static knowledge graph, where the triples are not being updated,

[1]https://www.snopes.com

[2]https://fullfact.org

is not sufficient to detect fake news at an early stage. Since news are often about recent events, including ones that are still in progress, sometimes the automatic knowledge-based fake news detection systems do not have enough data or knowledge about the ground truth. Similarly, to perform a manual fact-checking correctly, the human verifier needs to have access to the latest trustworthy information.

Therefore, it is beneficial to have a fake news detection system that can cope with that timeliness aspect of news articles. This project focused on building a system which utilises dynamic knowledge graph in storing the ground truth. More specifically, the system is able to gather new trusted article, extract the triples, and allow the addition or removal of the triples to and from the knowledge graph, making the knowledge graph dynamic. Furthermore, the system is also able to perform automatic fact-checking to a certain degree.

A further benefit of this knowledge-based method is that it is considered to be explainable, which means that it can be understood by humans. This explainability characteristic is leveraged in this project by shifting the focus of the system from fully automated to semi-automated. The system is intended to be used by trained human verifiers to help them detect fake news early by having the latest reliable facts in hand.

1.2 Aim and Objectives

As already discussed, this project aimed to develop a fake news detection system which uses a dynamic knowledge graph to store the ground truth in an attempt to help identify fake news at an early stage. In order to realise this aim, several objectives have been met in this project.

Firstly, an existing knowledge graph was chosen to be extended with the news articles data. Then, a triple extractor that extracts SPO triples from articles was developed. A mechanism which allows the existing knowledge graph to be extended by adding new triples and removing outdated triples was also built. Two algorithms to compare the to-be-verified facts with the dynamic knowledge graph were implemented.

It was mentioned that the focus of the system is to assist trained human verifiers. Therefore, it is crucial to have a user interface for the system which allows the human verifiers to do their job, which has been developed.

More interestingly, an evaluation to gauge the fact-checking performance of the system was also conducted.

1.3 Professional Considerations

This project has been undertaken in compliance with the BCS Code of Conduct[3]. With regards to BCS Code of Conduct 1.1., there was not any distressing news article used in this project. Regarding BCS Code of Conduct 1.2., the tools and libraries used in this project are free to use, if not open-sourced. Articles from the news sites that were used in the evaluation are allowed to be used in this project because this is a non-commercial project, as per the guidance from the British Intellectual Property Office[4]. Also equally important, I have taken university modules that are relevant to this project, such as Natural Language Engineering and Software Engineering, which means that I have a professional competence to do this project, as outlined in BCS Code of Conduct 2.1.

Additionally, there were no ethical issues encountered throughout the project. The data used came from news articles from trusted news sites, which means that they did not contain any personal information or distressing content. The usability testing was conducted in compliance with all of the points outlined in the Ethical Compliance Form, which is presented in Appendix C.

The rest of the report is structured as follows. The related background research is presented in Chapter 2. Then, Chapter 3 lists and discusses the different requirements for the system developed in this project. The aspects of the implementation itself, including the system architecture and the algorithms, are explained in detail in Chapter 4. Next, the modes of evaluation and their results are shown and discussed in Chapter 5, and finally the report is concluded in Chapter 6.

[3] https://www.bcs.org/membership/become-a-member/bcs-code-of-conduct/

[4] https://www.gov.uk/guidance/exceptions-to-copyright

Chapter 2

Background

In this chapter, several background research related to this project are presented. Firstly, some definitions of fake news are discussed, which includes the definition used in this project. Then, the concepts of semantic triple, knowledge graph, and dynamic knowledge graph are explained in detail. Those concepts are the backbone of this project. Equally important, automated and semi-automated fact-checking are discussed next.

2.1 Fake News

In developing a fake news detection system, it is necessary to decide what constitutes as fake news. In [9], fake news is broadly defined as "false news" and more narrowly defined as "intentionally false news published by a news outlet". In the literature, the term fake news is also often used alongside disinformation, misinformation, deceptive news, and rumour, which each has its own specific definition. However, for the purpose of this project, fake news is simply defined as "collection of non-factual information".

In relation to knowledge graph of true facts, however, there is a difference between information that is exactly known to be false and information that is unknown, which does not necessarily mean that it is false. Therefore, in this sense, what constitutes as fake news needs to be carefully defined.

2.2 Semantic Triple

Semantic triple is a data representation of a Subject-Predicate-Object (SPO) expression, which is why sometimes it is also referred as SPO triple. In linguistics, a subject in a sentence is the person or entity about whom the sentence is made, a predicate is the verb or property that conveys the action of the subject, and an object is a thing that is actioned

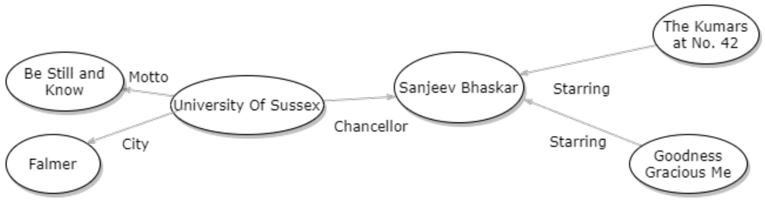

Figure 2.1: Example of a knowledge graph.

by the subject. For example, from the sentence “John Doe ignored social distancing.”, the subject is `John Doe`, the predicate is `ignored`, and the object is `social distancing`. Sometimes predicate is also called as relation because it shows the relation between subject and object.

In natural language processing, semantic triples are the outputs of Open Information Extraction (Open IE) task, which aims to extract relational triples from unstructured text. Traditionally, IE was done with a lot of human involvement in hand-crafting relation extraction rules and hand-tagging training examples that are domain-specific, but Open IE introduced a paradigm that allows a domain independent semantic triples extraction without requiring any human input [12]. Examples of such Open IE systems are Stanford's Open IE [13] and IIT's Open IE [14].

A more specific version of a semantic triple, even sometimes considered to be the same, is RDF (Resource Description Framework) triple. As defined in [15], in an RDF triple, each of the subject and predicate needs to be an IRI (Internationalized Resource Identifier), for example, `http://example.org/John_Doe`. The object, however, can be an IRI or a literal value, be it a string, integer, or other types of literal value. Not only that this RDF format is machine readable, the triple now can be queried unambiguously because of the unique identifier of its members.

2.3 Knowledge Graph

As previously described, a knowledge graph is a collection of SPO triples that is structured as a graph. The nodes in the graph represent the subject and object entities, while the edges connecting the nodes represent the relationships between the entities. Figure 2.1 shows an example of a small knowledge graph that contains knowledge of University of Sussex and Sanjeev Bhaskar. It can be seen that many kinds of inference can be drawn from the knowledge graph. For example, this query, “In which city is the university which has an actor of Goodness Gracious Me as its chancellor located?”, will return “Falmer” as the answer.

In this project, knowledge graph is used as the ground truth for the fact-checking process. It is mentioned that the knowledge graph is populated with triples that are extracted from trusted news articles. However, instead of building it from scratch using the news articles only, it is better to use a pre-existing knowledge graph to overcome the knowledge incompleteness that might come from it. The pre-existing knowledge graph can be extended with the news article triples.

Several pre-existing open knowledge graphs were discussed and compared in [16], namely YAGO, Wikidata, and DBpedia. YAGO is an open knowledge graph that gathers its information from Wikipedia and combines them with WordNet synsets. One strong point of YAGO is its high quality, which is proven by its manually verified accuracy of 95% [17]. Different to YAGO, Wikidata is a knowledge graph that is focused on being crowdsourced, which means that members of the public are allowed to extend and modify the knowledge inside it [18].

Lastly, DBpedia is static knowledge graph that has over than 21 billion triples extracted from Wikipedia with a monthly release cycle [19]. A dynamic version of DBpedia, named DBpedia Live was also developed. It is kept synchronised with Wikipedia with a delay of no more than a few minutes [20]. Note that the raw unstructured text from the Wikipedia article itself is not mainly used in the DBpedia extraction. DBpedia only extracts structured information that are mostly located in the infobox that is usually placed in the top right corner of a Wikipedia page [21].

As investigated in [16], none of the three open knowledge graphs can be considered as the best. Although all of them provide data set that is easily accessible from public or private endpoints and can be queried using SPARQL[5], none has the complete entities and relationships. However, in terms of timeliness, both Wikidata and DBpedia Live score highly because of the crowdsourcing nature of Wikidata and Wikipedia that is used by DBpedia Live.

2.4 Dynamic Knowledge Graph

Building a dynamic knowledge graph that can be updated automatically is deemed to be a non-trivial technical challenge. An attempt to solve this was presented in [22], which resulted in NOUS, an end-to-end framework to construct a dynamic knowledge graph. It combines an existing curated knowledge graph, such as YAGO, with knowledge extracted from unstructured text.

[5] https://www.w3.org/TR/rdf-sparql-query/

Although the knowledge graphs produced by NOUS are domain-specific, the steps taken by the framework can be used in a more general settings. New potential knowledge are gathered from various sources via news articles and web crawls and they are inputted to the framework in a streaming fashion. From the natural language text, triples are extracted using OpenIE, while named entity extraction and coreference resolution are also being performed. Then, the subjects and objects in the raw triples need to be matched to the entities in the existing knowledge graph. A new node or relation is created instead if the entities or relations are not found. It is acknowledged that mapping the relations is a challenge and still requires refinement in NOUS, due to the vast amount of relations produced by the triple extractor.

2.5 Automatic Fact-checking

Zafarani and Zhou [9] explained a general approach of a knowledge-based automatic fact-checking process in their survey. The process is divided into two stages, namely fact extraction (or knowledge base construction) and fact-checking itself. In the fact extraction stage, the knowledge or facts are extracted from the sources as raw facts which are then compiled to construct the knowledge graph. However, the raw facts needs to be cleaned-up beforehand by addressing some issues, including redundancy, invalidity, conflicts, unreliability, and incompleteness. The first four issues are worsened when multiple sources are used, but the incompleteness issue is alleviated by it.

In the fact-checking stage, an authenticity of a news article is done by comparing the extracted SPO `(Subject, Predicate, Object)` triples from the to-be-verified article with the true knowledge contained in the knowledge graph. When fact-checking a triple, firstly, the entities of the `Subject` and `Object` are matched with the nodes in the knowledge graph. If the nodes are found, the triple is considered as truth if there is an edge labelled as `Predicate` connecting the nodes. Otherwise, the triple's authenticity can be considered as false if a closed-world assumption is used. If that assumption is not used, further knowledge inference can be done to compute the possibility of it being true.

The fact-checking problem can be defined formally using the following formulae,

$$\mathcal{F} : (s_i, p_i, o_i) \xrightarrow{G_{KB}} A_i,$$

$$A = I(A_1, A_2, ..., A_n),$$

where F is a function that assigns the authenticity value A_i to each triple (s_i, p_i, o_i) in the to-be-verified article by comparing them to the knowledge graph G_{KB} and I is an aggregation function which aggregates all A_i's. The to-be-verified article is true if A is 1 and is completely false if A is 0.

2.6 Towards Automatic Fact-checking

It needs to be acknowledged that creating a fully-automated fact-checking system with a high accuracy is a difficult task. Therefore, in [23], a framework for a semi-automated fake news detection system was presented. The main idea of the proposed framework is to have a computer tool that is used interactively with an analyst who needs to ask the correct questions to the tool in order to analyse the validity of an information. Note that the domain used in that framework was specific to healthcare and the knowledge graph that is used did not consist of semantic triples, unlike what have been discussed in the previous sections. However, some of the ideas used in the framework are applicable to a more general use case.

The importance of the original selection of the knowledge graph content was highlighted because it affects the quality of the results and the analytical process done by the analyst. A good self-explanatory visualisation of the knowledge graph also seems to be useful as it was mentioned several times that the analyst had to centre their attention and navigate through the graph when performing the analysis. It was concluded that the framework could be the basis of other heuristic systems for analysis of dynamic information and a starting point for automatic fake news detection algorithms.

Chapter 3

Requirements Analysis

This chapter is dedicated to list the requirements, both functional and non-functional, of the system implemented in this project. The initial system design is briefly discussed beforehand.

3.1 Initial System Design

Figure 3.1 shows the initial high-level design of the system which consists of several components. Knowledge Graph Updater is the one responsible for keeping the knowledge graph updated by adding the triples extracted from news article. Therefore, it also needs to collect the news articles periodically and extract the triples from them.

As the name suggests, the fact-checker is the component that does the fact-checking. It extracts the triples from the submitted to-be-verified articles and queries the knowledge graph to perform the fact-checking.

The User Interface is a crucial component as it allows users to interact with the system. Users are able to submit an article and fact-check it, as well as adding new triples as feedbacks to the system.

3.2 Functional Requirements

3.2.1 Knowledge Graph Updater (KGU)

1. The KGU should mirror or update the knowledge graph if the existing open knowledge graph used is updated.

2. The KGU should scrape news articles from trusted news websites periodically, at least every 1 hour if the system is running all the time.

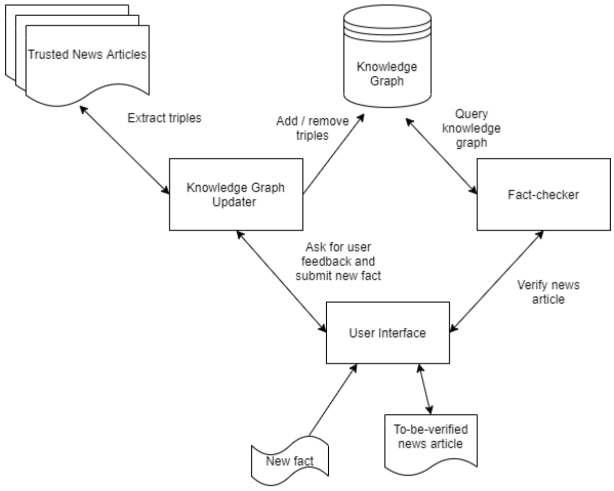

Figure 3.1: Initial system design.

3. The KGU shall extract SPO triples from the trusted news articles and from the user feedback.

4. The KGU shall be able to add the extracted triples to the knowledge graph if the triples do not exist yet in the knowledge graph.

5. The KGU shall be able to modify or remove triples from the knowledge graph if they are conflicting with the extracted triples.

3.2.2 Fact-checker

1. The fact-checker shall extract SPO triples from the to-be-verified news articles.

2. The fact-checker shall perform fact-checking algorithms on the to-be-verified triples.

3. Depending on the fact-checking algorithm, the fact-checker should be able to query the knowledge graph as needed.

4. The fact-checker shall return the calculated truthfulness score for the triples based on the fact-checking algorithms.

3.2.3 User Interface (UI)

1. In the fact-checking mode, the UI shall accept a news article in the form of sentences as input.

2. In the fact-checking mode, the UI should be able to accept a news article link as input.

3. In the fact-checking mode, the UI shall send the user input to the Fact-checker component.

4. In the fact-checking mode, the UI shall display the truthfulness score of an article received from the Fact-checker.

5. In the knowledge graph update mode, the UI shall accept triples as input.

6. In the knowledge graph update mode, the UI shall send the input to the KGU component.

7. In the knowledge graph update mode, the UI should return some form of feedback to the user stating that the knowledge graph has been updated, possibly by showing the related entities.

3.3 Non-Functional Requirements

1. The system shall be easy to use, at least in the fact-checking mode.

2. The system shall return the outputs to the user in real-time.

3. The system should be able to update the knowledge graph continuously in real-time.

4. The KGU and Fact-checker shall be written in Python as it offers plenty Natural Language Processing libraries.

5. The UI shall be a web interface written in HTML, CSS, and JavaScript.

Chapter 4

Implementation

This chapter explains in detail the implementation aspects of the system. After the implemented system architecture is discussed in the beginning, every component is explained one by one, starting with the DBpedia knowledge graph, web scraper, Triple Producer, Knowledge Graph Updater, Fact-checker, REST API, and the User Interface (UI). This chapter is ended by an extensive walkthrough of the UI to demonstrate what can be done with it.

4.1 System Architecture

The high-level architecture of the implemented system is presented in Figure 4.1. The components, that are slightly more defined than the ones presented in the initial design, are explained briefly in this section, before they are discussed in more detail in the following sections. The web scraper scrapes several trusted news websites and saves the scraped articles to the MongoDB database. The Knowledge Graph Updater's responsibilities are to extract the SPO triples from the scraped articles and add the triples to the DBpedia knowledge graph. They can be done automatically or on an on-demand basis triggered by the user's interaction. This component also accepts new triples to be added to or removed from DBpedia.

The Fact-checker checks whether a triple is true or not based on its existence in the knowledge graph and some other inferences. Two different algorithms can be chosen to be used. Other than triples, the Fact-checker also takes sentences and URL as inputs, where triple extraction and web scraping will be performed first in that case.

The User Interface (UI) is a crucial part in this system, as the aim of this system is to assist human verifiers. The UI, which is presented as a web app, allows user to perform

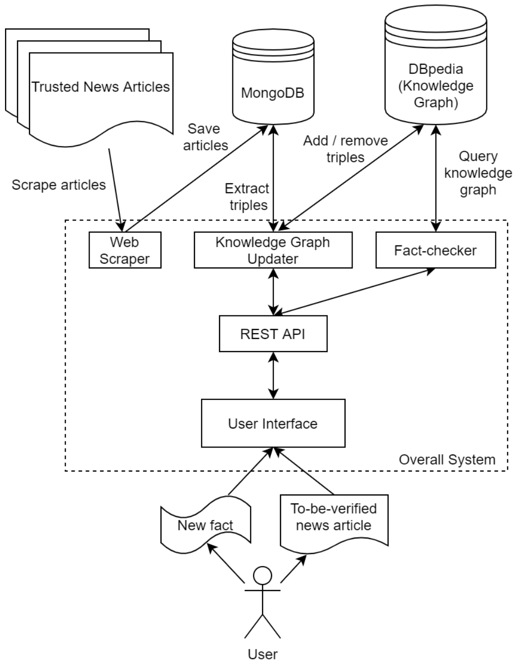

Figure 4.1: High-level system architecture.

fact-checking, update the knowledge graph, and look into the different entities that exist in the knowledge graph. The glue between the UI and the back-end components is the REST API, which is used by the UI and in turn calls the back-end components.

It was decided that all of the back-end components should be written in Python due to the vast amount of Natural Language Processing libraries that it offers. Meanwhile, the UI is written in React JS because it has a lot of open-source reusable components that can be adopted in a short amount of time. MongoDB[6] is chosen as the database to store the articles because it is document-oriented which makes it simple to use and known to a have a high performance. A more interesting decision was made when choosing DBpedia as the base knowledge graph, instead of YAGO or Wikidata. One of the main reason is that there is a possibility of having a continuously updated knowledge graph through DBpedia Live. The rule of thumbs presented in [24] to choose a knowledge graph was also considered, where in this case, DBpedia won because of its amount of existing relations and its interlinking to other knowledge graphs.

4.2 DBpedia

A small-scale version of DBpedia was mirrored and hosted locally in this project. A full version of the data dump was considered to be loaded, but the time it was taking to load

[6] https://www.mongodb.com/

was too long. Mirroring the DBpedia Live that always gets updated was also considered. However, in order to do that, the full version of the data dump needs to be loaded and the DBpedia Live synchronisation tool brings further complications. They are deemed to be not necessary for the current purpose of this project. Thus, only a small version of DBpedia is used here, although there is a possibility to expand it if the project is mature enough and ready for larger scale usage.

In DBpedia, the nodes, which represent Subjects and Objects, are called DBpedia entity resources. They are represented using URI in the form of http://dbpedia.org/resource/Entity_Name, such as http://dbpedia.org/resource/John_Doe and http://dbpedia.org/resource/Social_distancing. The edges connecting the nodes, which represent Relations or Predicates, are called DBpedia properties and they fit into the DBpedia Ontology. They are represented using URI in the form of http://dbpedia.org/ontology/propertyName, such as http://dbpedia.org/ontology/ignore.

To query DBpedia, a query language named SPARQL is used. Although it can be used to execute complex queries, a lot of boilerplates that is not necessarily related to knowledge graph semantics are required. A wrapper that only executes knowledge graph related operations is written to handle that.

4.3 Web Scraper

The web scraper scrapes articles from trusted news websites and saves them to the database. There are 3 trusted news websites that are scraped in this project, whose articles are used as the ground truth. They are BBC[7], The Independent[8], and The Guardian[9], which are chosen because of their timeliness, their content that are not provided behind paywalls, and their ease to be scraped. Generally, the scrapers are implemented using the Beautiful Soup 4[10] library. However, The Guardian Open Platform API[11] is used to get the articles from The Guardian, which allows for a better quality result, compared to scraping. If for some reason the article is not provided by the API, Beautiful Soup 4 is still used as a fallback. A scraped article is presented as a Python dictionary which has source (URL), date, headlines, and text as its properties.

When used by the Knowledge Graph Updater, the scraper also saves the article to

[7] https://www.bbc.co.uk/news
[8] https://www.independent.co.uk
[9] https://www.theguardian.com/uk
[10] https://www.crummy.com/software/BeautifulSoup/
[11] https://open-platform.theguardian.com/

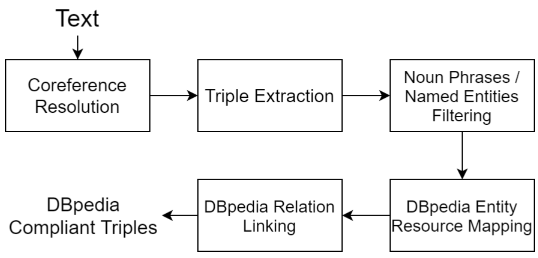

Figure 4.2: Triple Producer Pipeline.

MongoDB, which is not done if it is used by the Fact-checker when receiving a URL as input. Since the Fact-checker allows any URL as an input, a generic scraper was also implemented, whose purpose is merely to scrape any text on the web page.

As well as being used by the Knowledge Graph Updater on demand, the scraper can also run continuously in the background and scrapes latest news articles. This is possible by calling the RSS feed endpoint of each websites periodically, which contains a list of URLs of the latest news articles that are going to be scraped.

4.4 Triple Producer

Triple Producer is a module that is used to produce DBpedia compliant SPO triples from sentences. There are several operations that are performed in the pipeline before outputting the SPO triples, which is summarised in Figure 4.2. First, a coreference resolution is done to the text using Neuralcoref[12], which resolves coreference clusters using neural network. For example, "John Doe ignored social distancing. He was admitted to hospital on Sunday." would be transformed to "John Doe ignored social distancing. John Doe was admitted to hospital on Sunday."

Then, the raw SPO triples are extracted from each sentence. As mentioned in Section 2.2, Open IE systems can be used to extract the triples. In this module, two choices of Open IE system are provided, namely Stanford Open IE[13] and IIT Open IE[14] as introduced in [13] and [14], respectively. The Stanford Open IE is mainly used, however, because the system is more mature and produces a reasonable amount of triples. Using the running example, an example of the extracted triple would be `(John Doe, ignored, social distancing)`.

After the raw triples are extracted, an optional filtering is performed, whose pur-

[12] https://github.com/huggingface/neuralcoref

[13] https://stanfordnlp.github.io/CoreNLP/openie.html

[14] https://github.com/dair-iitd/OpenIE-standalone

pose is to only allow triples that has named entities or noun phrases as the Subject and Object. To decide if a piece of text is a named entity, noun phrase, or neither, Spacy[15] is used. Following the filtering, the Subject and Object need to be mapped to the resources that already exist in DBpedia. This is done using DBpedia Spotlight [25] by calling its REST API[16]. The extracted triple from the previous example would be changed to (`http://dbpedia.org/resource/John_Doe`, `ignored`, `http://dbpedia.org/resource/Social_distancing`). The Subject in the triple needs to be in a DBpedia URI format. Therefore, if the Subject was not mapped to a DBpedia resource by DBpedia Spotlight, it is simply converted into the appropriate format. An Object, however, is allowed to be a literal. Hence, it is only converted to the DBpedia URI format if the resource already exists in the local DBpedia.

The next step in the pipeline is to link the Relation to a DBpedia relation. A tool called Falcon [26] was considered to perform this relation linking. However, its performance is really slow and it is only suitable for relations between resources that already exist in DBpedia. Thus, it was quickly disregarded and the Relation is converted to a DBpedia format manually instead, by firstly performing lemmatisation using Spacy. For the previous example, the output of this step would be (`http://dbpedia.org/resource/John_Doe`, `http://dbpedia.org/property/ignore`, `http://dbpedia.org/resource/Social_distancing`). If for some reason, a component of the triple is empty, the triple is removed.

Essentially, all of the above steps are performed to the text, which in the end outputs a list of triples with their corresponding sentences.

4.5 Knowledge Graph Updater

Knowledge Graph Updater (KGU) is a component whose purpose is to carry out many operations related to updating DBpedia knowledge graph. It is also a component which users interact with through the UI. When run in the background, it constantly checks for article stored in MongoDB which does not have triples extracted from it yet. If there is one, KGU uses Triple Producer to produce the triples from the article and saves the triples in MongoDB. There is also an option to automatically insert non-conflicting triples to DBpedia. Here, a triple is said to have a conflict if there exists in the knowledge graph another triple with the same Subject and Relation as the ones of the triple.

Being the intermediary between the user and the knowledge graph, KGU is respons-

[15] https://spacy.io/

[16] https://www.dbpedia-spotlight.org/api

ible for retrieving extracted triples from articles, adding triples to the knowledge graph, removing triples from the knowledge graph, retrieving all triples related to an entity, and some other operations.

4.6 Fact-checker

As the name suggests, Fact-checker is a component that performs fact-checking. There are two types of Fact-checker that are implemented, which are Exact Match Fact-checker and Non-exact Match Fact-checker.

In the former, the fact-checking algorithm simply only cares about triples that exactly match the triples in the knowledge graph. Given a list of to-be-verified triples, for every triple, it checks if the exact same triple, with the same Subject, Relation, and Object, exists in the knowledge graph. If it does, the triple is marked as `exists`. It also checks for conflicts, in the same sense as explained in the previous section, which is a triple with the same Subject and Relation. If a to-be-verified triple is found to be conflicting with another triple, it is marked as `conflicts`. If no exact match and no conflict is found for the to-be-verified triple, it is marked as `none`, which means it is an unknown triple.

For the Non-exact Match Fact-checker, the fact-checking algorithm is a bit more complex. It still finds exact matches and conflicts as the previous algorithm. But, possible matches are also inferred in this algorithm. A to-be-verified triple is said to have possible matches, if at least one of the followings is fulfilled:

- There exists a triple in the knowledge graph with the opposite relation from the to-be-verified triple. Specifically, opposite means that the triple is in the form of (Object, Relation, Subject), instead of (Subject, Relation, Object). This is to accommodate triples that were extracted from passive sentences. For example, from the sentence "Social distancing was ignored by John Doe", the produced triple would be (`http://dbpedia.org/resource/Social_distancing`, `http://dbpedia.org/property/ignore`, `http://dbpedia.org/resource/John_Doe`).

- There exists a triple in the knowledge graph with the same Subject and Object as the ones in the to-be-verified triple and with a synonymous Relation. For example, (`http://dbpedia.org/resource/John_Doe`, `http://dbpedia.org/property/ignore`, `http://dbpedia.org/resource/Social_distancing`) already exists in the knowledge graph, while the to-be-verified triple is (`http://dbpedia.org/resource/John_Doe`, `http://dbpedia.org/property/neglect`, `http://dbpedia.org/resource/Social_` distancing). Since ignore and neglect are synonymous, this is a possible match. All synonyms of the Relation are retrieved using WordNet[17].

- There exists a triple in the knowledge graph with the same Subject and Object as the ones in the to-be-verified triple. One could argue that if there already exists a relation between a Subject and Object and the Relation is not synonymous, then it would be a conflict. However, that is not always the case, as both (http://dbpedia.org/resource/John_Doe, http://dbpedia.org/property/ignore, http://dbpedia.org/resource/Social_distancing) and (http://dbpedia.org/resource/John_Doe, http://dbpedia.org/property/hate, http://dbpedia.org/resource/Social_distancing) are valid triples that could exist at the same time, for example. A further comparison between the two relations should be made.

- There exists a triple in the knowledge graph with a corefering entity as the Subject or Object. If the submitted to-be-verified fact is in the form of text, the to-be-verified triples extracted from the text should contain triples with all corefering entities. For example, from the sentences "Mr John Doe ignored social distancing. John Doe was admitted to hospital", in addition to (http://dbpedia.org/resource/Mr_John_Doe, http://dbpedia.org/property/ignore, http://dbpedia.org/resource/Social_distancing), the triple (http://dbpedia.org/resource/John_Doe, http://dbpedia.org/property/ignore, http://dbpedia.org/resource/Social_distancing) is also fact-checked with the usual rules explained above. This is to alleviate any errors made by the coreference resolver when choosing the main corefering entity.

The output of the Fact-checkers is a list of triples where every triple is mapped to its result (exists, conflicts, possible, or none), with the sentences from which the triples are extracted.

4.7 REST API

The REST API acts as the interface between the UI, and the KGU and Fact-checker components. Here, it is implemented using Flask[18] because of its simplicity to create APIs quickly. The REST API has little to no logic included in it, as it delegates the responsibilities to the appropriate components.

[17] https://wordnet.princeton.edu/

[18] https://flask.palletsprojects.com/en/1.1.x/

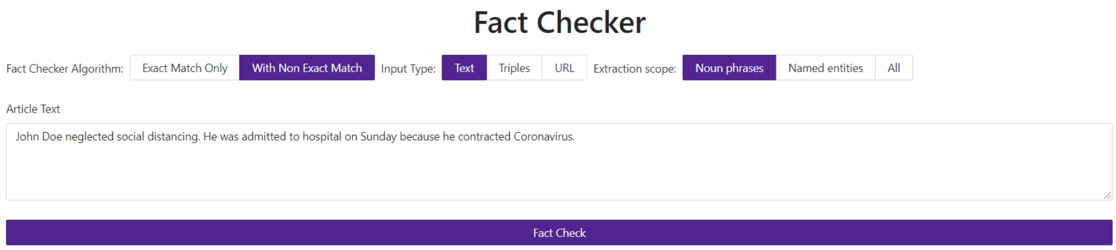

Figure 4.3: Fact-check view with text as input.

4.8 User Interface

4.8.1 Implementation Details

As previously mentioned, the UI is implemented as a web application using React JS due to its wide availability of free libraries that provide reusable components. Ant Design[19] is chosen to be the React UI library used in this project because of its easy adoption, customisable components, and detailed documentation. Perhaps, two most important Ant Design's components that are being used extensively in this project are Table[20] and Form[21].

The UI has no highly complicated logic as in general it only accepts user input, call the REST API endpoints, and present the data back to the user, with maybe some small data transformations to fit the UI format.

4.8.2 Walkthrough

This subsection is dedicated as a walkthrough that shows what users can do with this system using the UI. As the main purpose of this project is to assist human verifiers, it is important for the UI to be allowing them to do their job effectively. The UI is divided into several views or pages, based on the functionalities.

Fact-checker

In the Fact-checker view, users can submit to-be-verified knowledge, in the form of sentences, triples, or URL, as shown in Figure 4.3, Figure 4.4, and Figure 4.5, respectively. There is also the option to choose which fact-checking algorithm to use, either the Exact Match Only or the With Non Exact Match, as well as the option of triple extraction scope, either Noun phrases, Named entities, or All.

[19] https://ant.design/
[20] https://ant.design/components/table/
[21] https://ant.design/components/form/

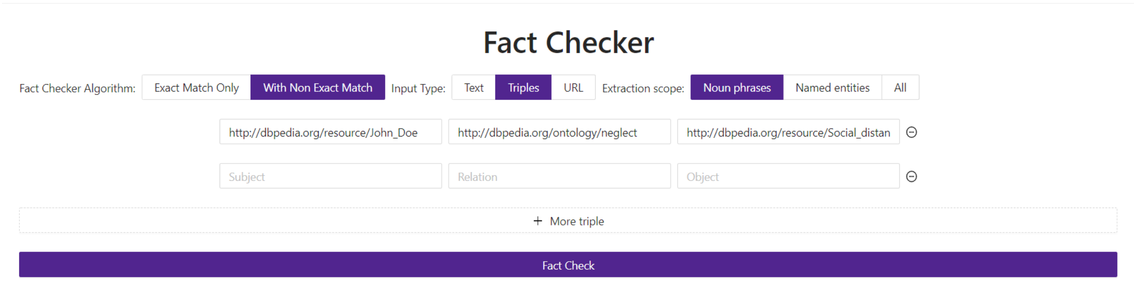

Figure 4.4: Fact-check view with triples as input.

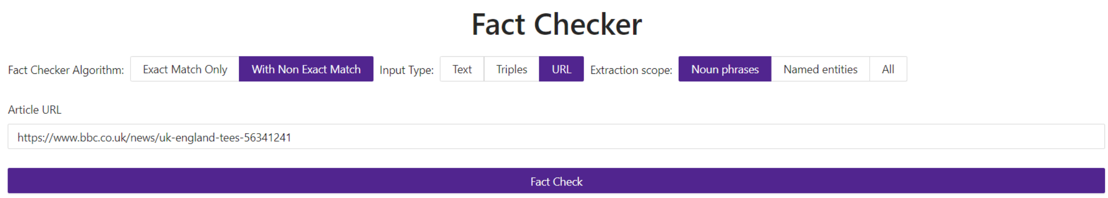

Figure 4.5: Fact-check view with URL as input.

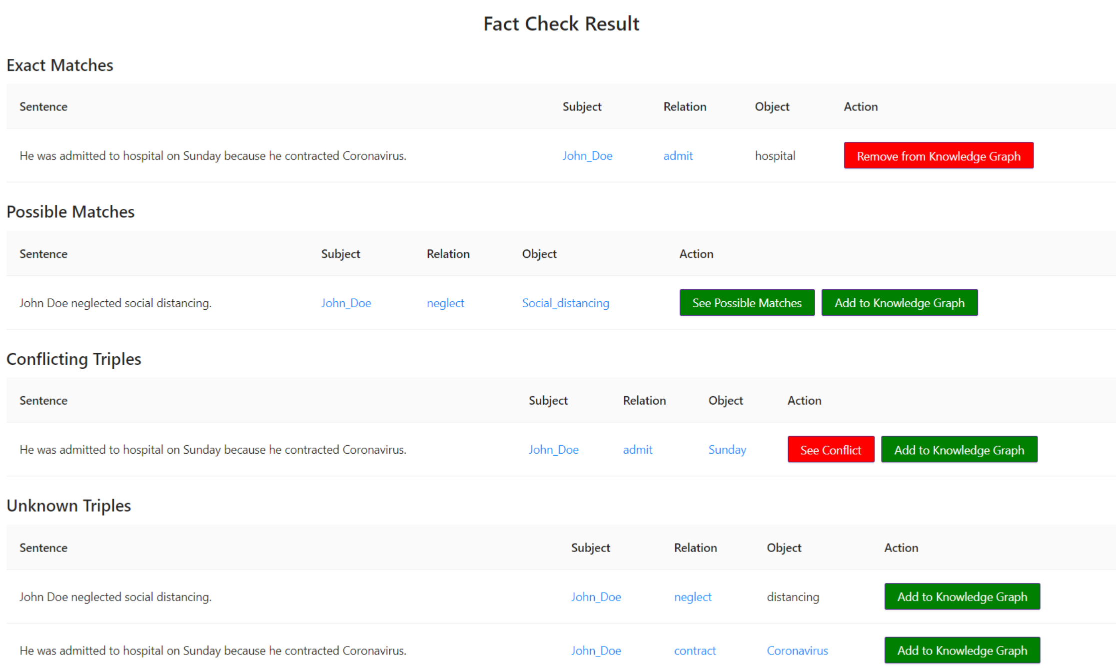

Figure 4.6: Fact-checking result using the non-exact match algorithm.

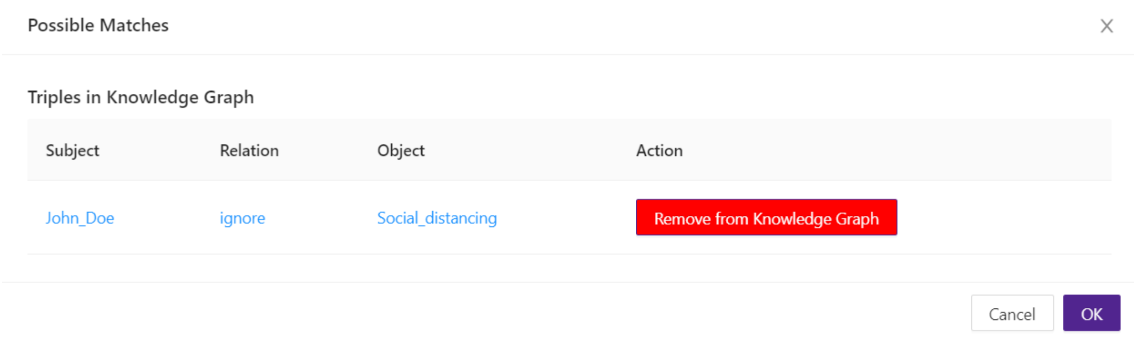

Figure 4.7: Possible matches modal.

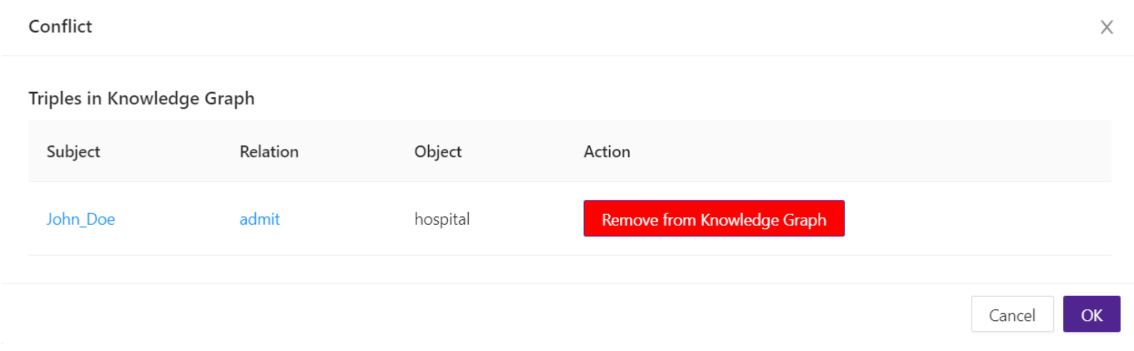

Figure 4.8: Conflicting triples modal.

An example of the fact-checking results are shown in Figure 4.6. The results are divided into 4 categories: Exact Matches (for triples marked as `exists` by the fact-checker), Possible Matches (for triples marked as `possible`), Conflicting Triples (for triples marked as `conflicts`), and Unknown Triples (for triples marked as `none`). As seen in the figure, every triple in each category has at least one button associated to the triple that users can click.

For triples in the Exact Matches category, users can click the button to remove individual triples from the knowledge graph. This might be needed if the human verifiers think that the knowledge is not valid anymore and has been superseded by a more recent knowledge. For triples in the Possible Matches category, there are two buttons that users can click, which are the button to see the possible matches and the button to add the triple to the knowledge graph. If the former button is clicked, it will open a modal, as shown in Figure 4.7, that lists the possible matches for the triple.

For triples in the Conflicting Triples category, there are also two buttons, including a button to see the conflicts and another to add the triple to the knowledge graph. When the first button is clicked, a modal will be opened, which lists the triples that are conflicting with the triple, as shown in Figure 4.8. For triples in the Unknown Triples category, the button is used to add the triple to the knowledge graph, which is necessary if the human verifiers found a knowledge that they know is valid but not yet added.

Note that at the end of the day, the users are the ones that decide whether a news is fake or not. This UI view, and the whole system, effectively, only show what is known to the knowledge graph. Perhaps having many conflicts is an indication of the news being fake, although users still need to verify if that is the case. In addition to facilitating the fact-checking process, this UI view at the same time also serves as a mean of updating the knowledge graph, which is done by clicking the buttons that add the triple to the knowledge graph or remove them.

Knowledge Graph Updater - Add New Article

In this UI view, users can add new articles that have not been scraped and added to the database before, as shown in Figure 4.9. Users can input the article's URL, choose the triple extraction scope, and pick whether the non-conflicting triples should be automatically added to the knowledge graph or not. After submitting the article, the users will get back a list of triples from the article that have not been added to the knowledge graph. Then, users can add them to the knowledge graph or discard them.

Knowledge Graph Updater - Add Own Knowledge

This view, shown in Figure 4.10, offers some auxiliary features for the users. They can add their own triples, that are not extracted from articles, to the knowledge graph. They can also make two DBpedia entity resources equal, which sometimes is useful when the Triple Producer's coreference resolution picked an incorrect entity as the Subject.

Knowledge Graph Updater - Pending Article Knowledge

This UI view is where users can interact with the scraped articles. As can be seen in Figure 4.11, users can click the Update button to trigger triple extraction for the scraped articles that have not had the triples extracted yet. They can choose to automatically add the non-conflicting extracted triples to the knowledge graph or not.

Users can also manually add the triples to the knowledge graph by choosing one of the articles listed in the table. After an article is chosen, a list of triples that has not been added to the knowledge is shown underneath, that is not unlike the list on the Add New Article view.

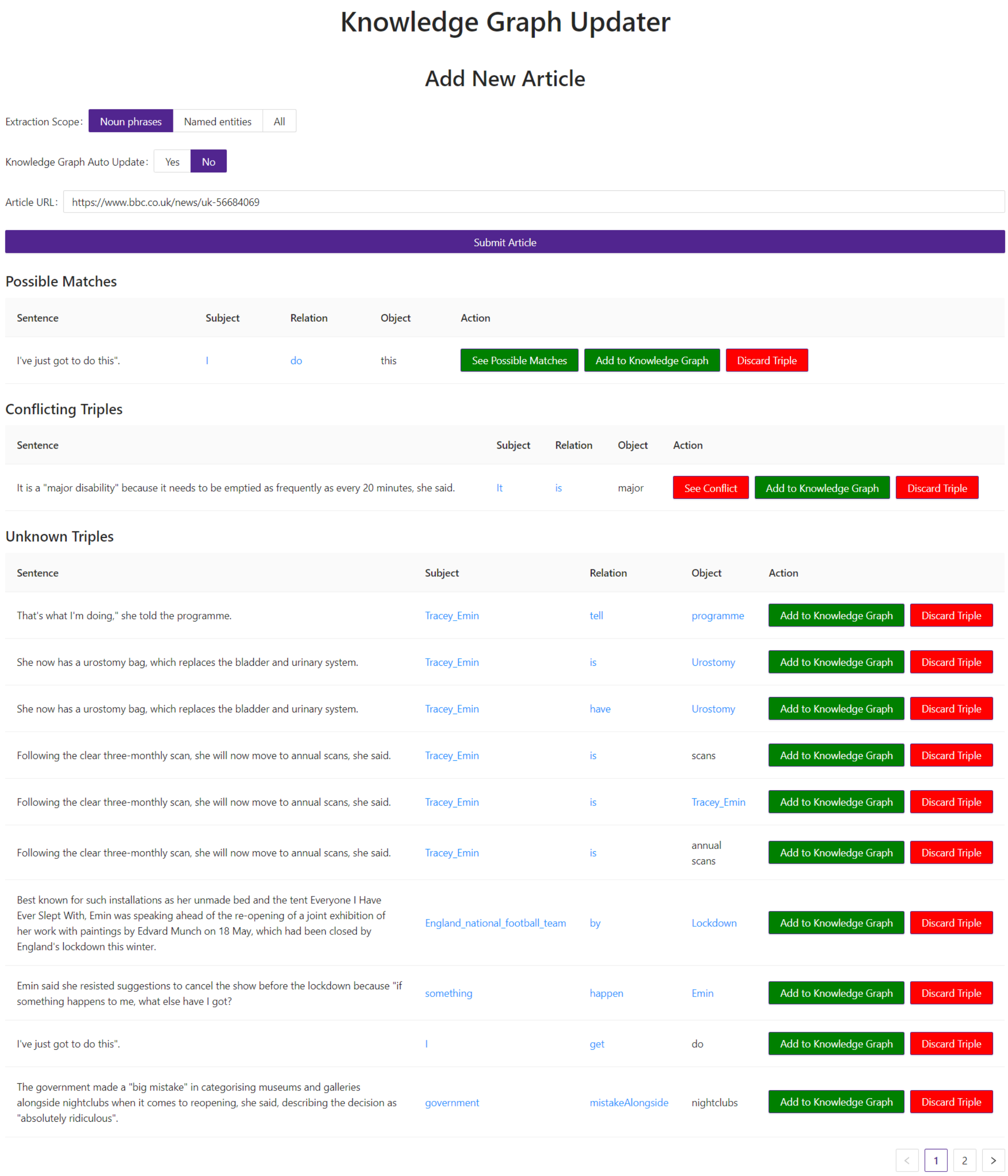

Figure 4.9: Add new article view.

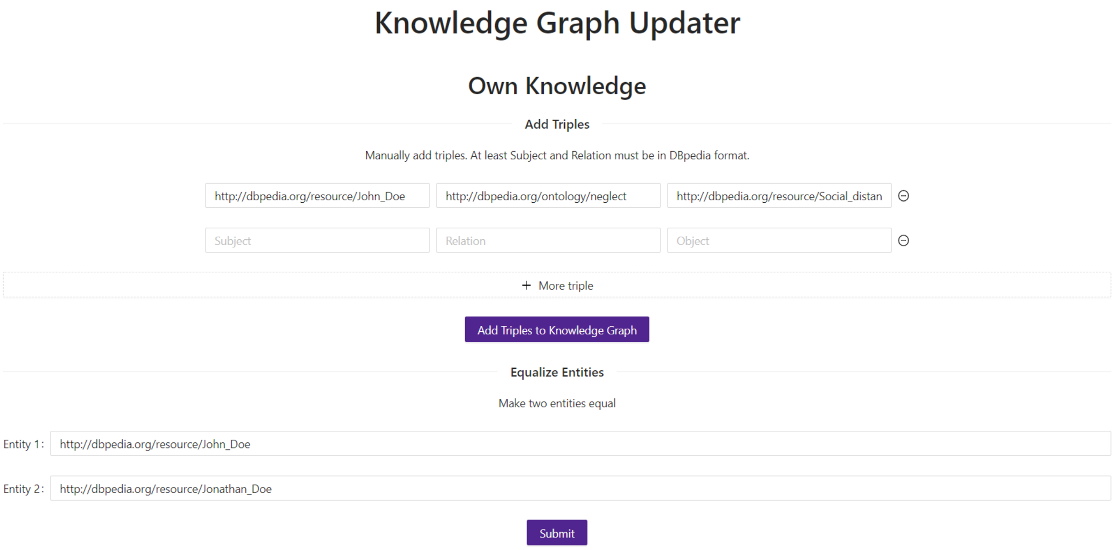

Figure 4.10: Add own knowledge view.

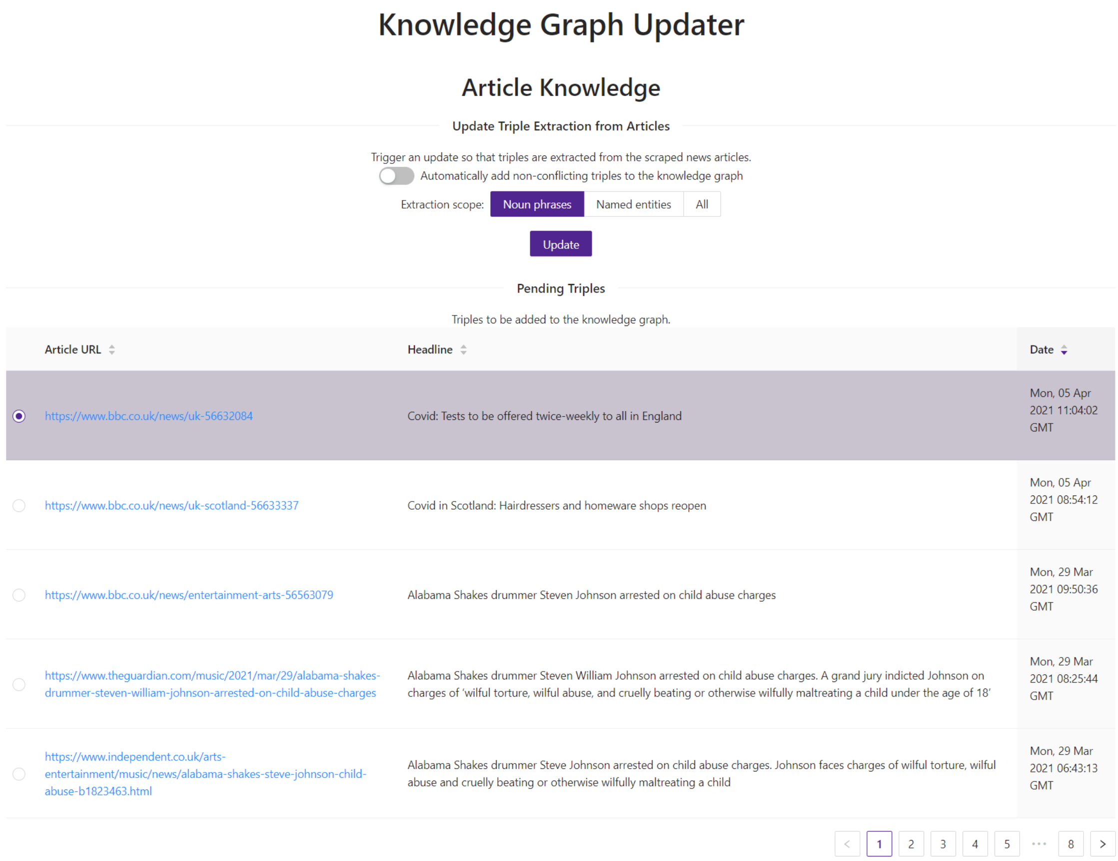

Figure 4.11: Pending article knowledge view.

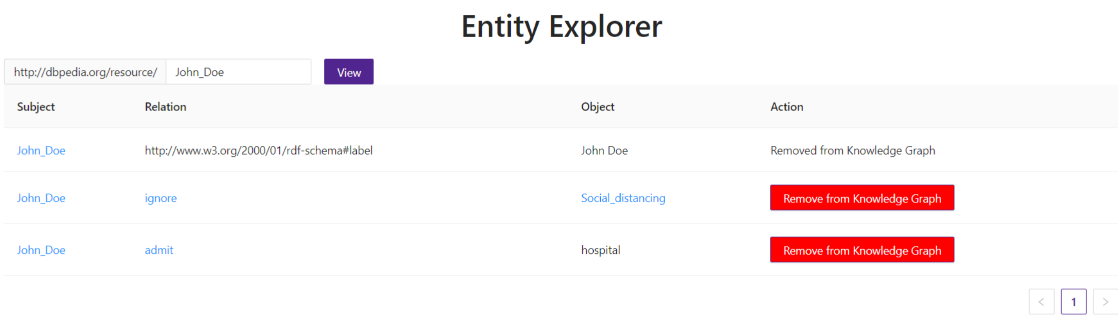

Figure 4.12: Entity explorer view.

Entity Explorer

This is an auxiliary view that allows users to look up an entity, see the triples related to it, and remove the triples, as shown in Figure 4.12. This is useful if users want to quickly amend an entity without having knowledge extracted from articles.

Chapter 5

Evaluation

In this chapter, the different methods of evaluation and their results are discussed in detail. First, an overview of the evaluation types and the reasons for doing them are explained. Then, the next three subsections describe how the evaluations were performed and their results in detail. The evaluations are to assess the triple quality, the fact-checking result, and the usability of the UI. Finally, evaluation of the system based on the number of requirements met is also presented.

5.1 Overview

This system was evaluated from four aspects. First, the quality of the triples extracted from news articles were analysed in relation to how they should have been extracted if the extraction was performed by human. A better quality triple means a better knowledge graph, which might lead to a better fact-checking system.

Second, the Non-exact Match Fact-checker algorithm's performance was evaluated. However, instead of feeding the fact-checker with fake news, it was used to fact-check real news. The reason behind this is because the default output of this fact-checker is to return `none` or unknown if the knowledge graph does not have any information mentioned in the to-be-verified news. Recall that the four states of fact-checking result in this system are `exists` or Exact Match (exact match is found), `conflicts` or Conflicting (a triple with the same Subject and Relation is found), `possible` or Possible Match (a possibly true triple is found, defined by the rules), and `none` or Unknown (no related triple is found). Evaluating by fact-checking fake news and expecting the result to be `conflicts` or `none`, while the default of the system is to output `none`, is not really useful. This is true especially when a lot of fake news is not direct opposite of what is reported in the real news. Therefore, it is more interesting to evaluate the fact-checker by inputting real news articles of the same topics from different sources and see how they are being considered as true fact or otherwise.

Third, the usability of the system and UI was also evaluated. This was done by asking 5 individuals to test the interface by completing some predefined tasks. This is quite important because the aim of this system is to assist trained human verifiers in performing fact-checking. Thus, the interface needs to be intuitive and easy enough to be used.

Fourth, the system is evaluated by simply looking at how many requirements are met and if the system is able to fulfil the requirements as expected. They also serve as the success criteria of this project.

It is important to note that the first two evaluations were done objectively, although the author did them solely and manually through the UI.

5.2 Triple Quality

To analyse and evaluate the produced triple quality, a total of 25 articles were submitted to the Knowledge Graph Updater via the Add New Article view of the UI. The articles were handpicked from 3 news websites, namely BBC, The Independent, and The Guardian, and they were chosen from the Entertainment & Arts domain, due to the amount of known named entities that occur in the domain. The chosen extraction scope was Noun Phrases, which means that only triples whose Subject and Object are noun phrases should be considered. This extraction scope was chosen because it produced a reasonable amount of triples, which is not too many and not too few, like what each All and Named Entities extraction scope would produce, respectively.

There is a total of 419 sentences collected from the 25 articles, but 122 sentences did not produce any triple. Out of the 122 sentences, there are at least 76 sentences that the author thinks should have triples extracted from them. Meanwhile, the rest, 46 sentences, do not contain appropriate triples with noun phrases, which means that they are expected to not produce triples. In other words, approximately 18.1% of the total sentences failed to produce triples that they should have had.

Further investigations could be made on why this is the case. Two possibilities immediately came to the author's mind. First, the triple extractor that is used, in this case, Stanford's Open IE, did not extract any triple at all from the sentence. It occurred to the majority of the cases, for example, "Newsbeat asked Mo Gilligan for comment, but

CATEGORIES	AMOUNT
Expected	**385**
Erroneous	**321**
Coreference resolution failed	126
Incorrect DBpedia mapping	81
Ambiguous/misleading meaning	4
Incorrect Subject	17
Incorrect Relation	38
Incorrect Object	24
Incorrect Triple	31
Total	**706**

Table 5.1: Categories of extracted triple quality

he has not responded." It is quite clear that one obvious triple should be extracted from the sentence, which is (`Newsbeat`, `asked`, `Mo Gilligan`), but it was not extracted. The reason behind why it was not extracted is not in the scope of this project, as the raw triple extraction depends on an external library.

Second, the extracted triple might have been filtered out during the filtering step. For example, from the sentence "In 2018, Anne Robinson was criticised", the triple extractor produced the raw triple (`Anne Robinson`, `was criticised in`, `2018`). However, in the noun phrases filtering step, this raw triple was filtered out, because it seems that `2018` was not considered as a noun phrase, which could be not difficult to fix.

The quality of the triples that did get produced from the sentences varies. Table 5.1 shows the different quality categories of the extracted triple along with the amount of triples. This categorisation was done manually by the author by looking at each produced triple, comparing it with the author's expected triple if the extraction was done by himself as a human, and finding the typical mistakes. From the 706 produced triples, 55%, or 385, of them are considered to be the same as what the author expected and have good quality. The rest contains mistakes which reduced their quality and correctness.

Most of them, 126 erroneous triples, were caused by the mistakes made in the coreference resolution, which is the first step in the Triple Producer pipeline. For instance, these two sentences, "Spike Lee to head Cannes Film Festival jury." and "The French Riviera festival on Tuesday announced that Lee will be president of the jury for the 74th Cannes.", each produced a raw triple where the Subject is `Spike Lee` and `Lee`, respectively. If the coreference resolution was working perfectly, the "Lee" in the second sentence should have been replaced with "Spike Lee", which would have resulted in a triple with `Spike Lee` as the subject. This would have led to a better triple collection of the `Spike Lee` entity. Note that the Neuralcoref's Spacy model used in this evaluation was the small model, which might explain why it made such mistakes. If a larger model was used, the performance of the coreference resolution would most likely be better.

The erroneous triples were also caused by incorrect DBpedia resource mappings that were done using DBpedia Spotlight. An example of such mistake is when Steven Johnson, who is a drummer of Alabama Shakes, was mapped to https://dbpedia.org/page/Steven_Johnson_(racing_driver) instead. This is not ideal, but it would have been accepted if it was consistent. However, in another text, the same Steven Johnson was mapped to a different DBpedia resource, namely https://dbpedia.org/page/Stephen_C._Johnson. This inconsistency clearly would affect the fact-checking process because the supposedly same information could occur in two different entities instead.

A minority of the erroneous triple has ambiguous or misleading meaning. For example, `(Sarah_Harding, die, "hospital")`, which was extracted from "... she nearly died of sepsis in hospital ...", has a different semantic from the intended meaning of the text. The rest of the problems are generally about Subjects, Relations, or Objects that the author thinks are wrong or could be better. The cause of these problems might be the triple extractor library being used (in this case, Stanford's OpenIE), as it tends to produce a lot of relationships with different phrases being used, such as `die` and `nearly die`.

As will be seen in the next section, the quality of the triples did contribute to the performance of the fact-checker algorithm.

5.3 Fact-checking Result

The Non-Exact Match Fact-checker algorithm was evaluated by submitting real news articles from different sources that are of the same topic and fact-checking them against each other. More specifically, there were 10 sets of articles that were used in this evaluation, where each set consists of articles of the same topic. Five sets contain articles from three different sources, namely BBC, The Independent, and The Guardian, while the other five contain articles only from two sources, which are BBC and The Independent only.

For each set, every article in the set was added to the knowledge graph via the Add New Article view of the UI one by one. After an article was added and consequently considered as the ground truth, the remaining articles in the set were fact-checked by submitting them via the Fact-checker view, with "With Non-Exact Match" as the fact-checker algorithm, "URL" as the input type, and "Noun Phrases" as the extraction scope. This was done in turn for every article in every set. Ideally, given that the knowledge graph has been fed some knowledge of the same topic, the fact-checker would find matches for most of the triples.

Additionally, for the five sets that have articles from three sources, three more fact-checkings were performed each. That is, by pairing each article in one set with each other and submitting them to the knowledge graph as the ground truth. Then, the one article that was not part of the pair was fact-checked.

Table 5.2 summarises how the fact-checking evaluation was performed, as explained above. For every fact-checking result, there are corresponding ideal results that the author thinks the fact-checker should have given if the process was performed by human. The ideal results were not attained only from the extracted triples, but also from the context and sentences where the triples were extracted from. This is because the quality of the extracted triples has affected the performance of the fact-checker and thus the ideal results should also assume that a perfect Triple Producer is used instead. The author gave an "Exact Match" as the ideal result to the triples who could have the same exact triple extracted from the ground truth articles, a "Possible Match" to triples whose semantics can also be found in the ground truth articles in any form, and an "Unknown" to triples that cannot be found in the ground truth articles. No "Conflicting" was given as an ideal result, because all articles used in this evaluation are real news, which means no conflicting information should be found, even though it might violate the definition of conflicting that was defined in Section 4.6. Furthermore, qualitative comments were added when necessary to show why the triples have the outputted results. Appendix B fully shows the fact-checking results, ideal results, and comments in detail.

Since the fact-checking results are put into four categories, for the purpose of evaluation, this task of fact-checking can be thought of as a multi-class classification. Table 5.3 and Table 5.4 show the confusion matrices of the fact-checking result, with one article and two articles as the ground truth, respectively. From the confusion matrices, the precision, recall, and F1-scores were calculated, which are summed up in Table 5.5.

It can be seen that the precisions for the "Exact Match" are 1, but it is less interesting because if an exact match was found, it does mean that there is indeed an exact match. The recalls are not as high, however, which are 0.52 and 0.54 for the 1 article and 2 articles as ground truth, respectively, which means that there are many triples that were supposed

Article(s) used as ground truth	**Fact-checked article**		
	BBC	The Independent	The Guardian
BBC		✓✓	✓
The Independent	✓✓		✓
The Guardian	✓	✓	
BBC & The Independent			✓
BBC & The Guardian		✓	
The Independent & The Guardian	✓		

Table 5.2: Fact-check evaluation for a set of articles. Two check marks in a cell indicates that the fact-checking scenario is done for sets whose articles are from two sources and three source. One check mark indicates that the scenario is performed for sets whose articles are from three sources only.

		Ideal Result			
		Exact Match	Possible Match	Conflicting	Unknown
Result	Exact Match	103	0	0	0
	Possible Match	10	48	0	3
	Conflicting	20	27	0	8
	Unknown	64	220	0	609

Table 5.3: The confusion matrix of the fact-checking treated as a multi-class classification task, when using only 1 article source as the ground truth.

		Ideal Result			
		Exact Match	Possible Match	Conflicting	Unknown
Result	Exact Match	59	0	0	0
	Possible Match	6	15	0	0
	Conflicting	11	13	0	3
	Unknown	34	92	0	173

Table 5.4: The confusion matrix of the fact-checking treated as a multi-class classification task, when using 2 article sources as the ground truth.

	1 Article as Ground Truth			2 Articles as Ground Truth		
	Precision	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**
Exact Match	1.00	0.52	0.69	1.00	0.54	0.70
Possible Match	0.79	0.16	0.27	0.71	0.12	0.21
Conflicting	0.00	0.00	0.00	0.00	0.00	0.00
Unknown	0.68	0.98	0.81	0.58	0.98	0.73

Table 5.5: Precision, recall, and F1-score of the fact-checking.

to have exact matches, but were not found. Most of the mistakes that led to this came from the inconsistency of the Triple Producer in extracting triples from direct quotes. A lot of articles in the same set tend to have same direct quotes included in the articles, which should lead to exact matches when they are fact-checked against each other. But, that is not the case because the Triple Producer does not consistently produce the same triples. That might have happened because sometimes the direct quotes are split into different sentences, or the coreference resolution modified the direct quotes.

The precisions and recalls of the "Conflicting" are irrelevant because the Ideal Results do not have any "Conflicting" triples in it, as explained previously.

The recalls of Unknown are high, at 0.98 for both, which means that the majority of the expected unknown triples are also found to be unknown. That is also not as interesting because the default of the system is to output Unknown and this is expected because articles from different sources, even if they are reporting on the same topic, often have additional information that is not presented by the other news sources.

On the rare occasions where the expected unknown triples are not found to be unknown, it happened because the relation of the triples are the same, which leads to having Conflicting as output, but actually the relation is a very general relation, such as `is` or `have` and that triple's information was actually not found in the ground truth.

On the other hand, the precisions of Unknown are not high, at 0.68 and 0.58, meaning that there is quite a lot of triples that have Unknown as the fact-check output while they are supposed to have Exact Match or Possible Match as output. The reasons for the ones that are meant to have Exact Match have been talked about earlier. Meanwhile, an investigation on the ones that are meant to have Possible Match are going to be discussed next, together with the precisions and recalls of the Possible Match class.

Firstly, the precisions of the Possible Match are 0.79 and 0.71, which can be considered quite good. That means that a lot of the triples that have Possible Match as results are expected to have Possible Match as per the Ideal Result. This was achieved by having the matching rules that try to find triples with the same Subject and Object. For example, the triple (http://dbpedia.org/resource/Tony_Bennett, http://dbpedia.org/property/diagnose, http://dbpedia.org/resource/Alzheimer) has a possible match with (http://dbpedia.org/resource/Tony_Bennett, http://dbpedia.org/property/have, http://dbpedia.org/resource/Alzheimer), which in the context of the article is correct because the semantic are very similar. This makes sense because usually, but not always, two entities only have one exclusive Relation, which leads to the assumption that triples that have the same Subject and Object are semantically close. This assumption needs to be treated carefully because there will be inevitably two triples with the same entities, but totally conflicting Relations. That is what should be considered as misinformation. Unfortunately, that has not been implemented yet in this current system. Other matching rules, such as synonymous relations and opposite relational structure (Object - Relation - Subject), have not been found to be useful in this evaluation process, although they might be if more experiments were performed.

Note that even though the precisions of the Possible Match are quite high, they are contrasted with the very low recalls, which are 0.16 and 0.12. It shows that there are many triples that are supposed to have Possible Match, but the outputs are Unknown or Conflicting instead. This is perhaps the most interesting part of the evaluation to be discussed, as it reveals the improvements to the fact-checking algorithm that could be made.

The cases where the triples were said to have Conflicting, instead of Possible Match, are caused by the rule that says that if two triples have the same Subjects and Relations, they should be considered as Conflicting, regardless of what the Relations are. This is not always the case, as have been mentioned before, because that will only be the case if the Relations are totally conflicting. Sometimes, triples with the same Relations are complementing each other, as opposed to conflicting each other. For instance, (http://dbpedia.org/resource/Clive_Myrie, http://dbpedia.org/property/replaceJohnHumphrys, "new host") is complementing (http://dbpedia.org/resource/Clive_Myrie, http://dbpedia.org/property/replaceJohnHumphrys, http://dbpedia.org/resource/Mastermind_(TV_series)) and not conflicting it. To overcome this issue, the definition of Conflicting in this system needs to be changed so that it is more rigid and clear.

The major cause of the low recalls are the triples that were outputting Unknown, rather than Possible Match. Most of them are because of the information in the triples are presented differently and the sentences that have the information did not produce any triples. This has been discussed in Section 5.2 and it shows how important for the Triple Producer to be of a high quality in order to have a good fact-checker. Although the performance of the fact-checker itself is also of importance, it cannot be denied that having the triples being extracted in the first place is fundamental. Without having the triples, there is almost no use of having a best-performing fact-checker algorithm.

There are also information that are presented implicitly in the triples that require some sophisticated inferences to decide if they are matching or not. For instance, there is a triple (`http://dbpedia.org/resource/Sister_Act`, `http://dbpedia.org/property/push`, `"next July"`) that was extracted from the sentence “Sister Act has been pushed back until next July” and the triple needs to be fact-checked. In the ground truth collection, there is another triple (`http://dbpedia.org/resource/Sister_Act`, `http://dbpedia.org/property/run`, `http://dbpedia.org/resource/19_July`) that was extracted from the sentence “Because of Covid restrictions, Sister Act will now run from 19 July to 28 August 2022”. From that, the truthfulness of the to-be-verified triple can be implied to be true. However, to come to that implication is not trivial.

Also related to the triple quality discussed in Section 5.2, the mistakes made in the coreference resolution affected the fact-checker performance. To give an example, from the sentence “The BBC newsreader Simon McCoy is leaving the corporation to join GB News”, this triple (`http://dbpedia.org/resource/McCoy`, `http://dbpedia.org/property/leave`, `"corporation"`) was extracted and it needs to be fact-checked. A good coreference resolver would have chosen `Simon_McCoy` instead of `McCoy`. Ideally, the `corporation` could also be resolved to `BBC`. If the coreference resolver was working perfectly, the fact-checker would have found a possible match or even an exact match for the triple, because there exists (`http://dbpedia.org/resource/Simon_Barlow`, `http://dbpedia.org/property/leave`, `http://dbpedia.org/resource/BBC_News`) in the knowledge graph.

Note that the last triple has `http://dbpedia.org/resource/Simon_Barlow`, not `http://dbpedia.org/resource/Simon_McCoy`, as the Subject. This shows how the DBpedia resource mapping step is also crucial for a good fact-checking performance. If the Subject was correctly mapped and the coreference resolution was better, that previous fact-checking would have given an Exact Match.

From all of these, it can be concluded that the quality of the triples produced does contribute massively to the performance of the fact-checker. Although the fact-checker algorithm seems to be working well so far, it did not have a chance to possibly discover more potential improvements, because the failure of Triple Producer to extract more triples has prevented that.

It might also worth noting that the performance of the fact-checker when using 1 article as ground truth does not differ much from when using 2 articles as ground truths, as can be seen again from Table 5.5. Surprisingly, the one with the lower performance is the one using 2 articles as ground truth, although not by much. Intuitively, having more articles should enrich the knowledge graph which could lead to a better fact-checking. However, that is not the case here, which is probably because only 5 sets of articles were used to do the experiment with 2 articles as ground truth. That means that there were only 15 article fact-checkings that were performed, as opposed to 40 (from 5 sets of 3 articles and 5 sets of 2 articles) for the experiment with only 1 article as ground truth. If more articles are submitted, the performance metrics should also be increasing.

5.4 Usability Testing

As the aim of the system is to assist trained human verifiers, it is quite important to make sure that the interface of the system allows the verifiers to do their job properly and easily. For that reason, usability testing was conducted, where each participant was asked to do several pre-defined tasks using the system's user interface. The tasks were designed to mimic the activities that the verifiers need to do on a daily basis. They include curating the good triples extracted from trusted articles, adding new trusted article, and fact-checking unverified article. The full list of task scenarios, along with the introduction and debriefing script, is presented in Appendix D.

Five individuals participated in this usability testing, where all of them are around the age of 21-22 years old. Four of them are male and one is female. The participants were encouraged to think out loud while doing the 10 tasks. The full results of the usability testing is presented in Appendix E.

Figure 5.1 shows the number of participants that managed to complete each task without any help. It can be seen that almost all of the tasks were completed successfully by all participants. The only exceptions are task 1 and task 10, where 2 participants failed to complete each of the tasks.

Task 1 is where the participants were asked to trigger the update that will extract triples from the collected articles. The 2 participants argued that they initially did not really understand the context which resulted in their failures to complete the task. How-

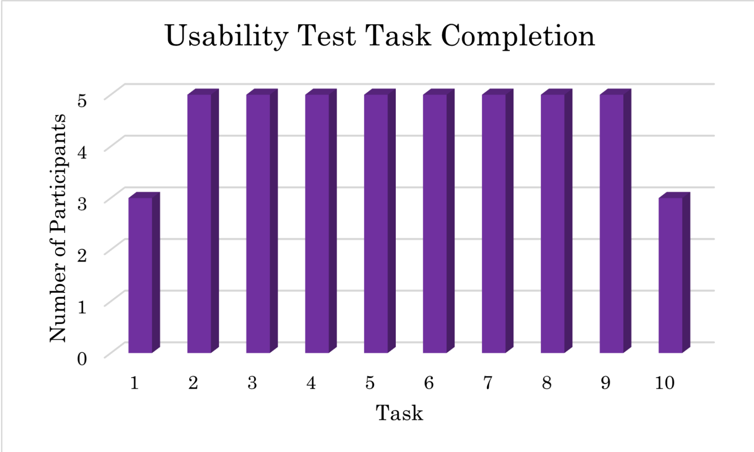

Figure 5.1: Usability testing task completion.

ever, after more explanation, they said that it actually made sense. For task 10, participants need to remove existing triple related to an entity. Two participants failed because they did not realise that they were able to remove triples from the Entity Explorer page.

Despite the failures by the minority of participants, all participants agreed that if the system is used by trained users, they will be able to use it with ease. Throughout the test, there were no system errors that were encountered. Most of the time, the participants were able to go to the correct page immediately without prior explanation, which implies that the system is intuitive to use. Many participants also provided suggestions on further improvement to the system. These are listed in Table 5.6, where the last improvement came from participants' failure on task 10 and the rest of improvements were explicitly suggested by the participants. Due to time restriction, only 8 improvements were able to be implemented. If there were more time, all improvements could have been implemented, which could enhance the user experience.

5.5 Requirements Completion

The system is also evaluated in terms of the number of requirements that are met. Appendix F shows the full results of the requirements completion. All 5 non-functional requirements were met successfully, while there is 1 functional requirement, out of 16, that was not met.

The failed requirement is "The KGU should mirror or update the knowledge graph if the existing open knowledge graph used is updated." which means if the base DBpedia is updated by the source provider, the local DBpedia should also be updated accordingly. This was not done in this project, because in order to dynamically mirror DBpedia, all

#	**Improvement**	**Action**
1	Show a notification after the triple extraction update process is done.	Implemented
2	Put a progress bar showing the progress of the triple extraction update process.	To Be Implemented
3	Allow selecting an article by clicking on the row, instead of only clicking on the radio button.	Implemented
4	Remove the confirmation modal and replace it with an in-place prompt for the add and discard triple buttons.	To Be Implemented
5	Put an undo button for the adding and removing triples operations.	To Be Implemented
6	Use larger buttons.	Implemented
7	Implement search, filter, and sort functionality for the triples table.	Implemented
8	Put a little help notice in Entity Explorer.	Implemented
9	Implement an auto-complete filling options based on the actual DBpedia content for the search input bar in Entity Explorer.	To Be Implemented
10	When there are empty triple lines in Add Own Knowledge, change it so the filled triples are added anyway without needing to remove the empty triples first.	To Be Implemented
11	In Add Own Knowledge, initially display the first triple input anyway without needing to click "More triples".	To Be Implemented
12	Put icons or symbols near each triples categories.	Implemented
13	Do not clear text in Fact-checker input when switching the input type from "text" to "URL".	Implemented
14	Add "Select All" or checkboxes to select multiple triples on triples table.	To Be Implemented
15	Add number per page option in Entity Explorer.	Implement
16	Change "Add Own Knowledge" page name.	To Be Implemented
17	Add another page separate from the Entity Explorer dedicated for removing triples.	To Be Implemented

Table 5.6: List of improvements based on usability testing.

of the datasets need to be loaded to the local DBpedia. This requires a lot of time and machine memory that the author's machine could not handle. As the system would still be working fine without this feature, it was decided to not spend much time to make this work.

Despite the failed requirement, the system still has fully working core functionalities and can be used without any issues, as also has been demonstrated in the usability testing.

Chapter 6

Conclusion

This project has shown how dynamic knowledge graph can be utilised as ground truth to detect fake news by performing fact-checking. As a framework, this system has introduced the steps that are needed to do the whole process of fact-checking, which were translated from the objectives of this project that have been achieved. DBpedia was chosen as the base knowledge graph, a Triple Producer that extracts and produces triples from articles was developed, and techniques to update the knowledge graph as well as the fact-checking algorithms were also implemented. As a system that aims to assist human verifiers and make the whole process explainable, there is a User Interface that has been developed.

An in-depth evaluation was done in terms of the performance of the fact-checker and the quality of the extracted triples. The evaluation has demonstrated how the triples quality is in line with the fact-checker performance, which means that the fact-checker could be better if the Triple Producer is made to be more accurate. This current fact-checker works relatively well when finding exact matches or concluding that an information is unknown, but a lot of improvements still could be made. Additionally, usability testing was also performed, which shows that the system is intuitive enough for novice users and will be easy to use for trained users.

With regards to future work, as has been mentioned many times before, the Triple Producer should be prioritised to be improved. This could be achieved by investigating other triple extraction methods and relation mapping techniques. Another avenue of triple extraction that could be explored is triple extraction using transformer architecture, such as the one presented in [27]. In the recent years, transformers have been the state of the art for other NLP tasks, which means that there is also a possibility to have a good triple extraction using the architecture. Meanwhile, mapping arbitrary relations to a predefined ontology is also still a difficult task that needs to be tackled.

Once the quality of triples has improved, a more automatic approach of updating the dynamic knowledge graph could be investigated. A simplistic approach would be to assume that if a new triple has a totally opposite relation with a triple in the knowledge, then it should go replace the triple, while other types of relations means that it is complementing. However, if possible, it might also be useful to have some sort of expiration timestamps on the triple when appropriate. This would help removing invalid knowledge automatically.

Finally, the fact-checker algorithm could be massively improved. A core improvement is to make better inference on whether two triples mean the same thing or not, linguistically. Perhaps some supervised approaches could be used to solve this, although explainability still needs to be considered. This is quite similar to paraphrase identification task in NLP, therefore inspirations could be taken from it. Approaching this issue from the logic or network flow perspective, similar to what has been done in [11], could also be another avenue worth trying.

Bibliography

[1] H. Allcott and M. Gentzkow, "Social Media and Fake News in the 2016 Election," *Journal of Economic Perspectives*, vol. 31, pp. 211–236, May 2017.

[2] G. Pennycook, J. McPhetres, Y. Zhang, J. G. Lu, and D. G. Rand, "Fighting COVID-19 Misinformation on Social Media: Experimental Evidence for a Scalable Accuracy-Nudge Intervention," *Psychological Science*, vol. 31, pp. 770–780, July 2020. Publisher: SAGE Publications Inc.

[3] B. A. Asaad and M. Erascu, "A Tool for Fake News Detection," in *2018 20th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, pp. 379–386, Sept. 2018.

[4] J. Z. Pan, S. Pavlova, C. Li, N. Li, Y. Li, and J. Liu, "Content Based Fake News Detection Using Knowledge Graphs," in *The Semantic Web – ISWC 2018* (D. Vrandečić, K. Bontcheva, M. C. Suárez-Figueroa, V. Presutti, I. Celino, M. Sabou, L.-A. Kaffee, and E. Simperl, eds.), vol. 11136, pp. 669–683, Cham: Springer International Publishing, 2018. Series Title: Lecture Notes in Computer Science.

[5] C. Castillo, M. Mendoza, and B. Poblete, "Information credibility on twitter," in *Proceedings of the 20th international conference on World wide web - WWW '11*, (Hyderabad, India), p. 675, ACM Press, 2011.

[6] X. Zhou and R. Zafarani, "Network-based Fake News Detection: A Pattern-driven Approach," *ACM SIGKDD Explorations Newsletter*, vol. 21, pp. 48–60, Nov. 2019.

[7] X. Zhou, A. Jain, V. V. Phoha, and R. Zafarani, "Fake News Early Detection: An Interdisciplinary Study," *arXiv:1904.11679 [cs]*, Sept. 2020. arXiv: 1904.11679.

[8] S. Castelo, T. Almeida, A. Elghafari, A. Santos, K. Pham, E. Nakamura, and J. Freire, "A Topic-Agnostic Approach for Identifying Fake News Pages," *Companion Proceed-*

ings of The 2019 World Wide Web Conference, pp. 975–980, May 2019. arXiv: 1905.00957.

[9] X. Zhou and R. Zafarani, “A Survey of Fake News: Fundamental Theories, Detection Methods, and Opportunities,” *ACM Computing Surveys*, vol. 53, pp. 1–40, Sept. 2020.

[10] S. S. Pavlova, “Using Knowledge Graphs for Fake News Detection,” BSc Dissertation, University of Aberdeen, Aberdeen, UK, 2018.

[11] P. Shiralkar, A. Flammini, F. Menczer, and G. L. Ciampaglia, “Finding Streams in Knowledge Graphs to Support Fact Checking,” in *2017 IEEE International Conference on Data Mining (ICDM)*, pp. 859–864, Nov. 2017. ISSN: 2374-8486.

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

[13] G. Angeli, M. J. Johnson Premkumar, and C. D. Manning, “Leveraging Linguistic Structure For Open Domain Information Extraction,” in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, (Beijing, China), pp. 344–354, Association for Computational Linguistics, 2015.

[14] S. Saha and Mausam, “Open Information Extraction from Conjunctive Sentences,” in *Proceedings of the 27th International Conference on Computational Linguistics*, (Santa Fe, New Mexico, USA), pp. 2288–2299, Association for Computational Linguistics, Aug. 2018.

[15] R. Cyganiak, D. Wood, and M. Lanthaler, “RDF 1.1 Concepts and Abstract Syntax.” W3C Recommendation, https://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/#section-rdf-graph, Feb. 2014.

[16] S. G. Pillai, L.-K. Soon, and S.-C. Haw, “Comparing DBpedia, Wikidata, and YAGO for Web Information Retrieval,” in *Intelligent and Interactive Computing* (V. Piuri, V. E. Balas, S. Borah, and S. S. Syed Ahmad, eds.), Lecture Notes in Networks and Systems, (Singapore), pp. 525–535, Springer, 2019.

[17] T. Pellissier Tanon, G. Weikum, and F. Suchanek, “YAGO 4: A Reason-able Knowledge Base,” in *The Semantic Web* (A. Harth, S. Kirrane, A.-C. Ngonga Ngomo,

H. Paulheim, A. Rula, A. L. Gentile, P. Haase, and M. Cochez, eds.), Lecture Notes in Computer Science, (Cham), pp. 583–596, Springer International Publishing, 2020.

[18] D. Vrandečić and M. Krötzsch, "Wikidata: a free collaborative knowledgebase," *Communications of the ACM*, vol. 57, pp. 78–85, Sept. 2014.

[19] M. Hofer, S. Hellmann, M. Dojchinovski, and J. Frey, "The New DBpedia Release Cycle: Increasing Agility and Efficiency in Knowledge Extraction Workflows," in *Semantic Systems. In the Era of Knowledge Graphs* (E. Blomqvist, P. Groth, V. de Boer, T. Pellegrini, M. Alam, T. Käfer, P. Kieseberg, S. Kirrane, A. Meroño-Peñuela, and H. J. Pandit, eds.), vol. 12378, pp. 1–18, Cham: Springer International Publishing, 2020. Series Title: Lecture Notes in Computer Science.

[20] J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P. N. Mendes, S. Hellmann, M. Morsey, P. van Kleef, S. Auer, and C. Bizer, "DBpedia – A large-scale, multilingual knowledge base extracted from Wikipedia," *Semantic Web*, vol. 6, no. 2, pp. 167–195, 2015.

[21] S. Hellmann, C. Stadler, J. Lehmann, and S. Auer, "DBpedia Live Extraction," in *On the Move to Meaningful Internet Systems: OTM 2009* (D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, R. Meersman, T. Dillon, and P. Herrero, eds.), vol. 5871, pp. 1209–1223, Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. Series Title: Lecture Notes in Computer Science.

[22] S. Choudhury, K. Agarwal, S. Purohit, B. Zhang, M. Pirrung, W. Smith, and M. Thomas, "NOUS: Construction and Querying of Dynamic Knowledge Graphs," in *2017 IEEE 33rd International Conference on Data Engineering (ICDE)*, pp. 1563–1565, Apr. 2017. ISSN: 2375-026X.

[23] P. Lara-Navarra, H. Falciani, E. A. Sánchez-Pérez, and A. Ferrer-Sapena, "Information Management in Healthcare and Environment: Towards an Automatic System for Fake News Detection," *International Journal of Environmental Research and Public Health*, vol. 17, p. 1066, Jan. 2020. Number: 3 Publisher: Multidisciplinary Digital Publishing Institute.

[24] M. Färber and A. Rettinger, "Which Knowledge Graph Is Best for Me?," *arXiv:1809.11099 [cs]*, Sept. 2018. arXiv: 1809.11099.

[25] P. N. Mendes, M. Jakob, A. García-Silva, and C. Bizer, "DBpedia spotlight: shedding light on the web of documents," in *Proceedings of the 7th International Conference on Semantic Systems - I-Semantics '11*, (Graz, Austria), pp. 1–8, ACM Press, 2011.

[26] A. Sakor, I. Onando Mulang', K. Singh, S. Shekarpour, M. Esther Vidal, J. Lehmann, and S. Auer, "Old is Gold: Linguistic Driven Approach for Entity and Relation Linking of Short Text," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, (Minneapolis, Minnesota), pp. 2336–2346, Association for Computational Linguistics, June 2019.

[27] H. Ye, N. Zhang, S. Deng, M. Chen, C. Tan, F. Huang, and H. Chen, "Contrastive Triple Extraction with Generative Transformer," *arXiv:2009.06207 [cs]*, Apr. 2021. arXiv: 2009.06207.

Appendix A

Code Listing

```
fake-news-detection
├── api/
├── articlescraper/
├── common/
├── docs/
│   ├── _build/html/
│   │   └── index.html
│   └── fakenewsdetection.pdf
├── evaluation-result/
│   ├── evaluation-result.xlsx
│   └── usability-test-result.xlsx
├── factcheckers/
├── knowledgegraphupdater/
├── ui/
├── .env.default
├── definitions.py
├── environment.yml
├── logger.conf
├── README.md
└── report.pdf
```

Below is a short explanation of the content of each directory and top-level file in the code repository:

- `api/`: Flask REST API
- `articlescraper/`: article scrapers and RSS pollers
- `common/`: common modules, such as Triple Producer and Knowledge Graph Wrapper
- `docs/`: python documentation of this package
- `docs/_build/html/index.html`: entry point of python package documentation in webpage format
- `docs/fakenewsdetection.pdf`: python package documentation in pdf format

- `evaluation-result`: evaluation results
- `evaluation-result/evaluation-result.xlsx`: spreadsheet containing evaluation result as discussed in Section 5.2 and 5.3. A sample of this evaluation result is presented in Appendix B
- `evaluation-result/usability-test-result.xlsx`: usability testing result as discussed in Section 5.4 and presented in Appendix E
- `factcheckers/`: exact match and non-exact match fact-checkers
- `knowledgegraphupdater/`: Knowledge Graph Updater and its runner
- `ui/`: all things related to the User Interface
- `.env.default`: default environment variables
- `definitions.py`: constants for logger
- `environment.yml`: Conda environment file containing list of external libraries
- `logger.conf`: logger config file
- `README.md`: project readme file
- `report.pdf`: this report

The code repository is also hosted at `https://github.com/albertus-andito/fake-news-detection`.

Appendix B

Evaluation Result

On the next page, a sample of the evaluation of the triple quality and fact-checking, as discussed in Section 5.2 and 5.3 respectively, is presented. The sample consists of only the result of 1 article evaluation. The full result of the evaluation can be seen in `evaluation-result.xlsx` file as explained in Appendix A.

							BBC			Independent		
#	**Sentence**	**Subject**	**Relation**	**Object**	**Triple Category**	**Triple quality comments**	**Result**	**Ideal Result**	**Comment**	**Result**	**Ideal Result**	**Comment**
	TOPIC 1: SARAH HARDING											
	BBC https://www.bbc.co.uk/news/entertainment-arts-56402388											
1	Sarah Harding, who has breast cancer, has revealed doctors have told her she won't see another Christmas.	doctors	tell	Sarah_Harding	Expected					Unknown	Possible match	If the Subject was "doctor", instead of "doctors", then this would have had a possible match (because of the opposite relation/O-R-S). However it does not always make sense to remove the plurality of an entity. Also, there is also the problem of the "doctor" object being a string literal, instead of a DBpedia resource.
2	The former Girls Aloud singer announced last year she had the disease, which had spread to other parts of her body.	former_Girls	have	"disease"	Coreference resolution failed	Subject didn't get coreference resolved. Object could have been resolved to breast cancer, also.				Unknown	Possible match	If the Subject was "Girls_Aloud", this sentence would have a similar semantic to "Girls_Aloud - diagnose - Breast_cancer", given that the Object could have been referred to "Breast_cancer".
3	"I am having a glass of wine or two during all this, because it helps me relax.	I	have	"glass"	Expected					Unknown	Unknown	No mention in the Independent article.
4	"I'm at a stage now where I don't know how many months I have left.	I	leave	"how many months"	Incorrect Relation	Wrong Relation				Unknown	Unknown	No mention in the Independent article.
5	"Who knows, maybe I'll surprise everyone, but that's how I'm looking at things."	I	maybeSurprise	"everyone"	Expected					Unknown	Unknown	No mention in the Independent article.
6	"Who knows, maybe I'll surprise everyone, but that's how I'm looking at things."	I	surprise	"everyone"	Expected					Unknown	Unknown	No mention in the Independent article.
7	Harding also revealed she had sepsis while she was being treated for cancer in hospital.	Sarah_Harding	have	Sepsis	Expected					Possible match	Possible match	This information was presented differently in the Independent article, as "she nearly died of sepsis." The triples extracted from that are (Sarah_Harding, die, Sepsis) and (Sarah_Harding, nearlyDie, Sepsis). This is a possible match because it involves the same Subject and Object.
8	Harding told the public about her cancer diagnosis in August last year.	Sarah_Harding	tell	"public"	Expected					Conflicting	Possible match	It is conflicting with (Sarah_Harding, tell, "doctor"). This information is presented differently in the Independent article, as "she announced", which could mean that she told the public.
9	She told The Times that due to the pandemic she had put off going to the doctor when she first started having symptoms.	Sarah_Harding	tell	Times	Expected					Conflicting	Possible match	It is conflicting with (Sarah_Harding, tell, "doctor"). This triple was failed to be extracted from the Independent article, where it was (Me, share, Times) instead. If the subject was extracted correctly, this could have been a possible match.
10	Now, she hopes that her story will encourage others to get themselves checked by a doctor if they have concerns.	others	have	"concerns"	Expected					Unknown	Unknown	No mention in the Independent article.
11	Now, she hopes that her story will encourage others to get themselves checked by a doctor if they have concerns.	others	check	"doctor"	Expected					Unknown	Unknown	No mention in the Independent article.
12	Harding shot to fame in 2002 as a contestant on Popstars: The Rivals - an ITV talent show which aimed to find both a new girl band and boy band.	Sarah_Harding	shoot	"contestant"	Ambiguous/misleading meaning	Triple is not correct semantically (not the expected meaning)				Unknown	Possible match	In the Independent article, this information was presented as "Rising to fame on Popstars: The Rivals…" The triples were not extracted, however.
13	Harding shot to fame in 2002 as a contestant on Popstars: The Rivals - an ITV talent show which aimed to find both a new girl band and boy band.	Sarah_Harding	shoot	"fame"	Expected					Unknown	Possible match	In the Independent article, this information was presented as "Rising to fame on Popstars: The Rivals…" The triples were not extracted, however.
14	She made it to the final and was voted into the group which became Girls Aloud, alongside Nicola Roberts, Nadine Coyle, Kimberley Walsh and Cheryl Cole (then Tweedy).	Sarah_Harding	vote	"group"	Expected					Unknown	Unknown	No mention in the Independent article.

#	Fact Checking Result with the following base articles											
	Guardian			BBC and Independent			BBC and Guardian			Independent and Guardian		
	Result	Ideal Result	Comment	Result	Ideal Result	Comment	Result	Ideal Result	Comment	Result	Ideal Result	Comment
	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
1	Unknown	Possible match	In the Guardian article, this information was presented as "she was told by a doctor", but the triple was not extracted. There is also the triple (doctor, tell, "me"). However, the Subjects are different, namely doctors and doctor. If the Subjects were the same, there should have been a conflict.	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	Unknown	Possible match	If the Subject was "doctor", instead of "doctors", then there would have been a possible match with the triple from Independent (Sarah_Harding, tell, "doctor") (because of the opposite relation/O-R-S) and a conflict with the triple from Guardian (doctor, tell, "me").
2	Unknown	Possible match	In the Guardian article, this information was presented differently as "she had been diagnosed with breast cancer", which has a similar meaning, but with different granularity.	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	Unknown	Possible match	See Independent and Guardian columns
3	Exact match	Exact match	Exact match but it is not particularly helpful because it is a direct quote, resulting in having I that could refer to many instances.	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	Exact match	Exact match	There is an exact match with the triple from Guardian article.
4	Exact match	Exact match	Exact match but it is not particularly helpful because it is a direct quote, resulting in having I that could refer to many instances.The triple itself (or the relation) doesn't really make much sense.	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	Exact match	Exact match	There is an exact match with the triple from Guardian article.
5	Unknown	Exact match	This triple was not extracted from the Guardian article.	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	Unknown	Exact match	See Independent and Guardian columns
6	Unknown	Exact match	This triple was not extracted from the Guardian article.	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	Unknown	Exact match	See Independent and Guardian columns
7	Exact match	Exact match	Good exact match.	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	Exact match	Exact match	There is an exact match with the triple from Guardian article. With the Independent article, it was only a possible match.
8	Unknown	Possible match	This information was presented differently as "The star revealed", where revealed could also mean to tell the public. Nevertheless, that triple was also not extracted.	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	Conflicting	Possible match	It is conflicting with (Sarah_Harding, tell, "doctor"). This information is presented differently in the Independent article, as "she announced",and in the Guardian article, as "the star revealed", which could mean that she told the public.
9	Unknown	Possible match	This information was presented with more granularity, which says that she told in her memoir that was published in the Times. However, that triple was not extracted.	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	Conflicting	Possible match	It is conflicting with (Sarah_Harding, tell, "doctor"). This triple was failed to be extracted from the Independent article, where it was (Me, share, Times) instead. If the subject was extracted correctly, this could have been a possible match.
10	Unknown	Unknown	No mention in the Guardian article.	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	Unknown	Unknown	See Independent and Guardian columns
11	Unknown	Unknown	No mention in the Guardian article.	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	Unknown	Unknown	See Independent and Guardian columns
12	Unknown	Possible match	This triple was not extracted from the Guardian article	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	Unknown	Possible match	See Independent and Guardian columns
13	Unknown	Possible match	This triple was not extracted from the Guardian article from the sentence "Harding, 38, shot to fame".	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	Unknown	Possible match	See Independent and Guardian columns
14	Unknown	Unknown	No mention in the Guardian article.	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	Unknown	Unknown	See Independent and Guardian columns

							BBC			Independent		
#	**Sentence**	**Subject**	**Relation**	**Object**	**Triple Category**	**Triple quality comments**	**Result**	**Ideal Result**	**Comment**	**Result**	**Ideal Result**	**Comment**
15	She made it to the final and was voted into the group which became Girls Aloud, alongside Nicola Roberts, Nadine Coyle, Kimberley Walsh and Cheryl Cole (then Tweedy).	Sarah_Harding	voteAlongside	Nicola_Roberts	Expected	Relation could be better	[redacted]			Unknown	Unknown	No mention in the Independent article.
16	She made it to the final and was voted into the group which became Girls Aloud, alongside Nicola Roberts, Nadine Coyle, Kimberley Walsh and Cheryl Cole (then Tweedy).	Sarah_Harding	make	"it"	Expected		[redacted]			Unknown	Unknown	No mention in the Independent article.
17	The group went on to have several UK hits, including Sound of the Underground, The Promise, Love Machine, Jump and Call The Shots.	group	have	UK_Singles_Chart	Coreference resolution failed	Subject could've been corefered as Girls Aloud	[redacted]			Unknown	Unknown	No mention in the Independent article.
18	Girls Aloud reunited in 2012 after a short hiatus, to release and tour a greatest hits album.	Girls	release	"hits album"	Incorrect Subject	Wrong Subject	[redacted]			Unknown	Unknown	No mention in the Independent article.
19	Girls Aloud reunited in 2012 after a short hiatus, to release and tour a greatest hits album.	Girls_Aloud	release	"hits album"	Expected		[redacted]			Unknown	Unknown	No mention in the Independent article.
20	Girls Aloud reunited in 2012 after a short hiatus, to release and tour a greatest hits album.	Girls	release	Greatest_hits_album	Incorrect Subject	Wrong Subject	[redacted]			Unknown	Unknown	No mention in the Independent article.
21	Girls Aloud reunited in 2012 after a short hiatus, to release and tour a greatest hits album.	Girls_Aloud	release	Greatest_hits_album	Expected		[redacted]			Unknown	Unknown	No mention in the Independent article.
22	Harding has since taken on several acting roles, including appearances in Run for Your Wife, and St. Trinian's 2: The Legend of Fritton's Gold.	Sarah_Harding	take	"several acting roles"	Expected		[redacted]			Unknown	Unknown	No mention in the Independent article.
23	Harding has since taken on several acting roles, including appearances in Run for Your Wife, and St. Trinian's 2: The Legend of Fritton's Gold.	appearances	in	Run	Incorrect Object	Wrong Object. Uninformative triple.	[redacted]			Unknown	Unknown	No mention in the Independent article.
24	Harding has since taken on several acting roles, including appearances in Run for Your Wife, and St. Trinian's 2: The Legend of Fritton's Gold.	Fritton_(near_Great_Yarmouth)	of	Gold	Incorrect Relation	Wrong Relation. Uninformative triple.	[redacted]			Unknown	Unknown	No mention in the Independent article.
25	Harding has since taken on several acting roles, including appearances in Run for Your Wife, and St. Trinian's 2: The Legend of Fritton's Gold.	Sarah_Harding	take	"roles"	Expected		[redacted]			Unknown	Unknown	No mention in the Independent article.
26	Harding has since taken on several acting roles, including appearances in Run for Your Wife, and St. Trinian's 2: The Legend of Fritton's Gold.	Sarah_Harding	take	"acting roles"	Expected		[redacted]			Unknown	Unknown	No mention in the Independent article.
27	In 2017, she won Celebrity Big Brother.	Sarah_Harding	win	Celebrity_Big_Brother_(British_TV_Series)	Expected		[redacted]			Exact match	Exact match	The exact same sentence is also in the Independent article.
28	Breast cancer is the most common cancer in the UK, with women over 50 more at risk than the under 40s.	Breast_cancer	commonCancer	"women"	Expected		[redacted]			Unknown	Possible match	No mention in the Independent article.
29	Breast cancer is the most common cancer in the UK, with women over 50 more at risk than the under 40s.	Breast_cancer	commonCancer	UK_Singles_Chart	Incorrect DBpedia mapping	Wrong Object. Possibly because coreference failed to be resolved.	[redacted]			Unknown	Unknown	Wrong Object. Possibly because coreference failed to be resolved.
30	Breast cancer is the most common cancer in the UK, with women over 50 more at risk than the under 40s.	Cancer	in	UK_Singles_Chart	Incorrect DBpedia mapping	Wrong Object. Possibly because coreference failed to be resolved.	[redacted]			Unknown	Unknown	No mention in the Independent article.
31	Breast cancer is the most common cancer in the UK, with women over 50 more at risk than the under 40s.	Breast_cancer	is	Cancer	Expected		[redacted]			Unknown	Unknown	No mention in the Independent article.
32	Breast cancer is the most common cancer in the UK, with women over 50 more at risk than the under 40s.	Breast_cancer	cancer	UK_Singles_Chart	Incorrect DBpedia mapping	Uninformative relation. Wrong Object. Possibly because coreference failed to be resolved.	[redacted]			Unknown	Unknown	No mention in the Independent article.
33	Breast cancer is the most common cancer in the UK, with women over 50 more at risk than the under 40s.	Cancer	is	"women"	Expected	Uninformative relation/triple	[redacted]			Unknown	Unknown	No mention in the Independent article.
34	Breast cancer is the most common cancer in the UK, with women over 50 more at risk than the under 40s.	Breast_cancer	cancer	"women"	Incorrect Relation	Wrong relation	[redacted]			Unknown	Unknown	No mention in the Independent article.
35	But there are many other factors that can increase a person's risk, including a family history of cancer and being overweight.	person	have	"risk"	Expected	Uninformative relation/triple	[redacted]			Unknown	Unknown	No mention in the Independent article.
36	Many treatments are available for breast cancer and survival is generally good if the disease is detected early.	Many_treatments	available	Breast_cancer	Expected		[redacted]			Unknown	Unknown	No mention in the Independent article.
37	Many treatments are available for breast cancer and survival is generally good if the disease is detected early.	treatments	available	Breast_cancer	Expected		[redacted]			Unknown	Unknown	No mention in the Independent article.

	Fact Checking Result with the following base articles											
	Guardian			BBC and Independent			BBC and Guardian			Independent and Guardian		
#	Result	Ideal Result	Comment	Result	Ideal Result	Comment	Result	Ideal Result	Comment	Result	Ideal Result	Comment
15	Unknown	Unknown	No mention in the Guardian article.	[redacted]						Unknown	Unknown	See Independent and Guardian columns
16	Unknown	Unknown	No mention in the Guardian article.	[redacted]						Unknown	Unknown	See Independent and Guardian columns
17	Unknown	Possible match	This information was presented differently as "Their debut single, Sound of the Underground, was the Christmas No 1 that year, and the first of four chart-topping singles during their career.", but the triple was not extracted.	[redacted]						Unknown	Possible match	See Independent and Guardian columns
18	Unknown	Unknown	No mention in the Guardian article.	[redacted]						Unknown	Unknown	See Independent and Guardian columns
19	Unknown	Unknown	No mention in the Guardian article.	[redacted]						Unknown	Unknown	See Independent and Guardian columns
20	Unknown	Unknown	No mention in the Guardian article.	[redacted]						Unknown	Unknown	See Independent and Guardian columns
21	Unknown	Unknown	No mention in the Guardian article.	[redacted]						Unknown	Unknown	See Independent and Guardian columns
22	Conflicting	Possible match	It is conflicting with (Sarah_Holding, take, "term role"). This triple is a better quality.	[redacted]						Conflicting	Possible match	It is conflicting with (Sarah_Holding, take, "term role") from Guardian article. This triple is a better quality.
23	Unknown	Unknown	No mention in the Guardian article.	[redacted]						Unknown	Unknown	See Independent and Guardian columns
24	Unknown	Unknown	No mention in the Guardian article.	[redacted]						Unknown	Unknown	See Independent and Guardian columns
25	Conflicting	Possible match	It is conflicting with (Sarah_Holding, take, "term role"). This triple is a better quality.	[redacted]						Conflicting	Possible match	It is conflicting with (Sarah_Holding, take, "term role") from Guardian article. This triple is a better quality.
26	Conflicting	Possible match	It is conflicting with (Sarah_Holding, take, "term role"). This triple is a better quality.	[redacted]						Conflicting	Possible match	It is conflicting with (Sarah_Holding, take, "term role") from Guardian article. This triple is a better quality.
27	Conflicting	Possible match	It is conflicting with (Sarah_Holding, win, "series") and (Sarah_Holding, win, "20th series"). This triple is a better quality.	[redacted]						Exact match	Exact match	There is an exact match with the triple from Independent article. Supposedly, there is a conflict with the triples from Guardian article, but an exact match is more prioritised.
28	Unknown	Unknown	No mention in the Guardian article.	[redacted]						Unknown	Unknown	See Independent and Guardian columns
29	Unknown	Unknown	No mention in the Guardian article.	[redacted]						Unknown	Unknown	See Independent and Guardian columns
30	Unknown	Unknown	No mention in the Guardian article.	[redacted]						Unknown	Unknown	See Independent and Guardian columns
31	Unknown	Unknown	No mention in the Guardian article.	[redacted]						Unknown	Unknown	See Independent and Guardian columns
32	Unknown	Unknown	No mention in the Guardian article.	[redacted]						Unknown	Unknown	See Independent and Guardian columns
33	Unknown	Unknown	No mention in the Guardian article.	[redacted]						Unknown	Unknown	See Independent and Guardian columns
34	Unknown	Unknown	No mention in the Guardian article.	[redacted]						Unknown	Unknown	See Independent and Guardian columns
35	Unknown	Unknown	No mention in the Guardian article.	[redacted]						Unknown	Unknown	See Independent and Guardian columns
36	Unknown	Unknown	No mention in the Guardian article.	[redacted]						Unknown	Unknown	See Independent and Guardian columns
37	Unknown	Unknown	No mention in the Guardian article.	[redacted]						Unknown	Unknown	See Independent and Guardian columns

#	Sentence	Subject	Relation	Object	Triple Category	Triple quality comments	BBC			Independent		
							Result	Ideal Result	Comment	Result	Ideal Result	Comment
38	If you have a story suggestion email entertainment.news@bbc.co.uk.	you	have	“story suggestion”	Expected		[redacted]			Unknown	Unknown	No mention in the Independent article.
	Unique Sentences	**19**			**Expected**	26				1	1	**Exact match**
	Sentences without triples	**9**			**Coreference resolution failed**	2				1	8	**Possible match**
	Sentences without triples that should have had triples	**8**			**Incorrect Subject**	2				2	0	**Conflicting**
					Incorrect Relation	3				34	29	**Unknown**
					Incorrect Object	1				38	38	**Total**
					Incorrect DBpedia mapping	3						
					Ambiguous/misleading meaning	1						
					Incorrect Triple	0						

	Fact Checking Result with the following base articles											
	Guardian			BBC and Independent			BBC and Guardian			Independent and Guardian		
#	Result	Ideal Result	Comment	Result	Ideal Result	Comment	Result	Ideal Result	Comment	Result	Ideal Result	Comment
38	Unknown	Unknown	No mention in the Guardian article.	[redacted]						Unknown	Unknown	See Independent and Guardian columns
	3	5	**Exact match**	[redacted]						4	6	**Exact match**
	0	11	**Possible match**	[redacted]						0	10	**Possible match**
	4	0	**Conflicting**	[redacted]						5	0	**Conflicting**
	31	22	**Unknown**	[redacted]						29	22	**Unknown**
	38	38	**Total**	[redacted]						38	38	**Total**

	Sentences without triples being extracted:	Possible ideal extracted triple		
		Subject	Relation	Object
1	An extract from her memoir, printed in The Times, said: "In December my doctor told me that the upcoming Christmas would probably be my last."	my_doctor	tell	me
2	She said she didn't want an exact prognosis, just "comfort" and to be "pain-free".	Sarah_Harding	notWant	exact_prognosis
3	Harding said in her memoir, Hear Me Out, she is "trying to live and enjoy every second of my life, however long it might be".			
4	She said she was put into an induced coma and placed on a ventilator.	Sarah_Harding	put	induced_coma
5	"Even once I was off the ventilator, I couldn't speak properly," she said. "All I could do was make noises that sounded like a chimpanzee trying to communicate."	I	off	ventilator
6	She said she had been "inundated" with support since then, including from her former Girls Aloud bandmates, adding: "I'm grateful beyond words for that."	Sarah_Harding	inundated	support
7	She appeared on the Channel 4 reality show The Jump in 2016, but was forced to pull out after suffering a knee injury.	Sarah_Harding	appear	The_Jump
8	The most common sign of breast cancer is a lump or thickening in the breast - but there are other symptoms too.	breast_cancer	sign	lump
9	These symptoms can be caused by other conditions, so it is important to get any lumps or changes checked by a doctor.	symptoms	cause	other_conditions

Appendix C

Ethical Compliance Form

Ethical Compliance Form for UG and PGT Projects*
School of Engineering and Informatics
University of Sussex

This form should be used in conjunction with the document entitled "Research Ethics Guidance for UG and PGT Projects".

Prior to conducting your project, you and your supervisor will have discussed the ethical implications of your research. If it was determined that your proposed project would comply with **all** of the points in this form, then both you and your supervisor should complete and sign the form on page 3, and submit the signed copy with your final project report/dissertation.

If this is not the case, you should refer back to the "Research Ethics Guidance for UG and PGT Projects" document for further guidance.

---

1. Participants were not exposed to any risks greater than those encountered in their normal working life.
   Investigators have a responsibility to protect participants from physical, mental and emotional harm during the investigation. The risk of harm must be no greater than in ordinary life. Areas of potential risk that require ethical approval include, but are not limited to, investigations that require participant mobility (e.g. walking, running, use of public transport), unusual or repetitive activity or movement, physical hazards or discomfort, emotional distress, use of sensory deprivation (e.g. ear plugs or blindfolds), sensitive topics (e.g. sexual activity, drug use, political behaviour, ethnicity) or those which might induce discomfort, stress or anxiety (e.g. violent video games), bright or flashing lights, loud or disorienting noises, smell, taste, vibration, or force feedback.

2. The study materials were paper-based, or comprised software running on standard hardware.
   Participants should not be exposed to any risks associated with the use of non-standard equipment: anything other than pen-and-paper, standard PCs, mobile phones, and tablet computers is considered non-standard.

3. All participants explicitly stated that they agreed to take part, and that their data could be used in the project.
   Participants cannot take part in the study without their knowledge or consent (i.e. no covert observation). Covert observation, deception or withholding information are deemed to be high risk and require ethical approval through the relevant C-REC.

---

*This checklist was originally developed by Professor Steven Brewster at the University of Glasgow, and modified by Dr Judith Good for use at the University of Sussex with his permission.

If the results of the evaluation are likely to be used beyond the term of the project (for example, the software is to be deployed, the data is to be published or there are future secondary uses of the data), then it will be necessary to obtain signed consent from each participant. Otherwise, verbal consent is sufficient, and should be explicitly requested in the introductory script (see Appendix 1).

4. No incentives were offered to the participants.
   The payment of participants must not be used to induce them to risk harm beyond that which they risk without payment in their normal lifestyle. People volunteering to participate in research may be compensated financially e.g. for reasonable travel expenses. Payments made to individuals must not be so large as to induce individuals to risk harm beyond that which they would usually undertake.

5. No information about the evaluation or materials was intentionally withheld from the participants.
   Withholding information from participants or misleading them is unacceptable without justifiable reasons for doing so. Any projects requiring deception (for example, only telling participants of the true purpose of the study afterwards so as not to influence their behaviour) are deemed high risk and require approval from the relevant C-REC.

6. No participant was under the age of 18.
   Any studies involving children or young people are deemed to be high risk and require ethical approval through the relevant C-REC.

7. No participant had a disability or impairment that may have limited their understanding or communication or capacity to consent.
   Projects involving participants with disabilities are deemed to be high risk and require ethical approval from the relevant C-REC.

8. Neither I nor my supervisor are in a position of authority or influence over any of the participants.
   A position of authority or influence over any participant must not be allowed to pressurise participants to take part in, or remain in, any study.

9. All participants were informed that they could withdraw at any time.
   All participants have the right to withdraw at any time during the investigation. They should be told this in the introductory script (see Appendix 1).

10. All participants have been informed of my contact details, and the contact details of my supervisor.
   All participants must be able to contact the investigator and/or the supervisor after the investigation. They should be given contact details for both student and supervisor as part of the debriefing.

11. The evaluation was described in detail with all of the participants at the beginning of the session, and participants were fully debriefed at the end of the session. All participants were given the opportunity to ask questions at both the beginning and end of the session.
    Participants must be provided with sufficient information prior to starting the session, and in the debriefing, to enable them to understand the nature of the investigation.

12. All the data collected from the participants is stored securely, and in an anonymous form.
    All participant data (hard-copy and soft-copy) should be stored securely (i.e. locked filing cabinets for hard copy, password protected computer for electronic data), and in an anonymised form.

Project title: Using Dynamic Knowledge Graph for Fake News Early Detection

Student's Name: Albertus Andito

Student's Registration Number: 21802720

Student's Signature:

Date: 10 November 2020

Supervisor's Name: Julie Weeds

Supervisor's Signature: JWeeds

Date: 10 November 2020

Appendix D

Usability Testing Script

Introduction Script

Hello, thank you for agreeing to participate in this usability test. The purpose of this is to evaluate the interface of my final year project implementation in terms of its usability. Just to let you know, you are allowed to withdraw from this test at any time if you want to do so.

Before we start, would it be okay for me to record this conversation, including the audio and the video? The recording will be used to further investigate the data that is collected throughout this usability test. After that is done, the recording will be deleted.

Firstly, for demographic purposes, can I ask your age and gender?

Now, I will give you a background on my project. The title of the project is "Using Dynamic Knowledge Graph for Fake News Detection", and one of the deliverables of the project is a web app. The aim for the web app is to help trained users to do fact-checking and verify if a news article is fake or not. The fact-checking in the system relies on the concept that a fact is represented as a semantic triple. A triple consists of a subject, relation/predicate, and object. For example, from the sentence "John Doe ignored social distancing", a triple can be extracted, which is "John Doe" as the subject, "ignore" as the relation, and "social distancing" as the object. If the triple exists in the knowledge base or database, then the fact is said to be true. In this project, the true triples are collected from news websites, such as BBC, Independent, and Guardian.

I admit that the algorithm is far from perfect, which is why the aim of the system is to assist trained human verifiers, instead of doing all of the fact-checking automatically. Therefore, the purpose of this usability test is not really to test the accuracy or the performance of the algorithm. Rather, it is to test the usability of the interface and how useful the design is when aiding a human.

The role of the human verifier is to look through the articles that have been collected and curate the triples that are extracted from the articles to be added to the knowledge base. The verifier can also fact-check articles that have not been verified.

In this usability test, I will share my screen and give you the control of my screen. Then, you will be asked to perform several tasks on the web app that I will open on my screen. You are encouraged to think out loud while you are doing the tasks. You can also give some comments throughout the test. Note that you are not being tested. Instead, the one being tested is this interface.

Do you have any questions so far? You are also welcome to ask questions any time during the test. However, if there is no question now, we shall start.

Tasks scenarios

1. There are a lot of trusted news articles that were collected overnight. However, the triples have not been extracted from the article. Trigger the update so that triples are extracted from the articles, but do not add the triples to the knowledge graph just yet.

2. Look at the first pending article with the triples extracted from it. Decide on several good quality triples and add them to the knowledge graph, while removing some triples that are of bad quality.

3. You feel like there is not enough information about that topic from that article, so you want to add a new trusted article. Add this article: `https://www.bbc.co.uk/news/world-us-canada-56948665`, and choose some good quality triples to be added to the knowledge graph. They are (`Joe_Rogan`, `deny`, `accussations`) and (`Joe_Rogan`, `sign`, `deal`).

4. You want to see everything that is currently known about Joe Rogan.

5. Add a new triple with `Joe Rogan` as the subject, `believe` as the relation, and `Vaccine` as the object.

6. Fact check the following text “Joe Rogan denied the accusations. He signed the autograph.” with exact matches only algorithm.

7. Look into the conflicting triple, add it to the knowledge graph and remove the existing conflicted triple.

8. Fact check the following article: https://albertus-andito.com/fake-news-example with non-exact matches algorithm.

9. Add the unknown triples to the knowledge graph.

10. Remove the triple (`Joe_Rogan`, `genre`, `Observational_comedy`).

Debriefing Script

That is all the tasks that I wanted you to do. I have several follow-up questions for you to answer. Other than what you have said already, is there anything specific that you want to comment on? Do you think that the interface is good for users who are trained to use it? Is there any other suggestion?

That is all of the questions I wanted to ask you. Do you have any question for me instead?

Thank you again for participating in this usability testing. Your input is valuable for this project.

Appendix E

Usability Testing Result

On the next page, the result of usability testing, as discussed in Section 5.4, is presented. It includes the task completion, author's observation, and comments made by the five participants.

#	Task/Question		Responses					Summary
			User 1	User 2	User 3	User 4	User 5	
			Age: 21, Gender: Female	Age: 21, Gender: Male	Age: 22, Gender: Male	Age: 21, Gender: Male	Age: 21, Gender: Male	
1	Trigger the update so that triples are extracted from the collected articles.	**Pass/Fail**	Fail	Fail	Pass	Pass	Pass	Only 3 out of 5 participants passed this task. However, this is likely due to the users not having a lot of context about the system is known at the beginning.
		Observation	User was confused on what needs to be done. She had to be given a hint.	User was confused on what needs to be done. He had to be given a hint.	User clicked on other pages before going to the correct page.	User took some time before deciding to go to the correct page.	User immediately went to the correct page.	
		Comments	User was not sure on what needs to be done at first.	User said that he did not fully understand the context at the beginning, but after being given the hint, it all made sense.	-	User said that the update process is straightforward.	User appreciated that the "Update" button becomes clickable again once the update process is done, but suggested that there is also a notification stating that it is done. User also suggested to have a progress bar stating how many articles have been extracted so far.	
2	Look at the first pending article with the triples extracted from it. Add some good quality triples and remove some bad quality triples.	**Pass/Fail**	Pass	Pass	Pass	Pass	Pass	All 5 participants passed this task. The article link was found to be quite confusing, but the radio button was intuitive enough.
		Observation	User took some time, before mistakenly clicked on the article link, but then immediately realised that she should have clicked the radio button.	User almost clicked the article link, but immediately decided to just click on the radio button.	User mistakenly clicked on the article link, but then immediately realised that he should have clicked the radio button.	User mistakenly clicked on the article link, but then immediately realised that he should have clicked the radio button.	User immediately noticed that he should not clicked on the link and clicked on the radio button instead.	
		Comments		-	User liked that there is a good error handling when trying to add malformed triple.	User suggested that an undo button would be useful in case a triple was mistakenly added or discarded.	User suggested to also allow clicking on the row instead of the radio button for choosing the article. User suggested to remove the confimation modal and replace it with an in-place prompt for the add and discard triple buttons.	
3	Add a new trusted article: https://www.bbc.co.uk/news/world-us-canada-56948665, and add some triples to the knowledge graph. They are (Joe_Rogan, deny, accusations) and (Joe_Rogan, sign, deal).	**Pass/Fail**	Pass	Pass	Pass	Pass	Pass	All 5 participants passed this task without any issue.
		Observation	User immediately went to the correct page.	User immediately went to the correct page.	User thought that he had to open the link instead of adding the article link to the system. But, he found the correct menu quite immediately.	User immediately went to the correct page.	User immediately went to the correct page.	
		Comments	-	-	User would like to see larger buttons because they are not using all of the spaces efficiently.	User said that the UI very self-explanatory and minimalistic. User suggested search functionality for the triples table.	User suggested search, filter, and sort functionalities for the triples table.	
		Pass/Fail	Pass	Pass	Pass	Pass	Pass	

		Observation	User was completely confused on what needs to be done and took a very long time before finally clicked the Joe Rogan text. The expectation was to use the Entity Explorer, but this was also acceptable.	User clicked on "Joe Rogan" instead of using the Entity Explorer, but that is also acceptable.	User immediately went to the correct page.	User took some time before deciding to go to the correct page.	User immediately noticed that Joe Rogaan is an entity and went directly to the Entity Explorer page. User tried to submit the entity name without using the DBpedia format.	
4	See everything that is currently known about Joe Rogan.	**Comments**	She said that she did not immediately clicked the Joe Rogan text because she thought it was related to the relation.	-	-	User said it was straightforward and intuitive, but admitted that the page name "Entity Explorer" is not the most helpful name.	User suggested a little notice or help saying that it needs to use DBpedia format next to the search input bar. User suggested auto-complete filling options based on the actual DBpedia content for the search input bar.	All 5 participants passed this task, with some participants taking some time to find out how to do it.
		Pass/Fail	Pass	Pass	Pass	Pass	Pass	
		Observation	User did not immediately go the correct page.	User immediately went to the correct page, but thought that he had to immediately click the "Add triples" button.	User took some time before realising he needed to go its own page.	User immediately went to the correct page.	User immediately went to the correct page.	
5	Add a new triple with Joe Rogan as the subject, believe as the relation, and Vaccine as the object.	**Comments**	-	-	-	User liked that the UI does not allow empty triple to be added to the knowledge graph, but suggested to add the other triples anyway instead of waiting until the empty triple line is deleted or filled.	User suggested that the first triple input to be initially shown without having to click "More triple".	All 5 participants passed this task.
		Pass/Fail	Pass	Pass	Pass	Pass	Pass	
		Observation	User immediately went to the correct page.	User initially mistakenly switched the input type to "URL" but then realised his mistake.	User immediately went to the correct page.	User immediately went to the correct page.	User immediately went to the correct page.	
6	Fact check the following text "Joe Rogan denied the accusations. He signed the autograph." with exact matches only algorithm.	**Comments**	-	-	-	-	User would like to see some icons or symbol indicating that the exact matches are true triples.	All 5 participants passed this task without any issue.
		Pass/Fail	Pass	Pass	Pass	Pass	Pass	
		Observation	User took some time before clicking the "See conflict" button.	User immediately knew that he had to click the "See Conflict" button.	User took some time before clicking the "See conflict" button.	User took some time before clicking the "See conflict" button.	User immediately knew that he had to click on the "See Conflict" button to see the conflicted triple and to remove it.	
7	Look into the conflicting triple, add it to the knowledge graph, and remove the existing conflicted triple.	**Comments**	-	-	-	-	-	All 5 participants passed this task.
		Pass/Fail	Pass	Pass	Pass	Pass	Pass	
		Observation	User immediately switched the input type to "URL" before inputting the url.	User put the url into the text input before switching to the "URL" input type.	User immediately switched the input type to "URL" before inputting the url.	User put the url into the text input before switching to the "URL" input type.	User put the url into the text input before immediately switching to the "URL" input type.	All 5 participants

8	Fact check the following article: https://albertus-andito.com/fake-news-example with non-exact matches algorithm.	**Comments**	-	User did not necessarily like that the text in the input disappeared when he switched the input type to "URL".	-	-	User did not necessarily like that the text in the input disappeared when he switched the input type to "URL".	All 5 participants passed this task, but some found some issue with the disappearing text when switching input type to URL.
9	Add the unknown triples to the knowledge graph.	**Pass/Fail**	Pass	Pass	Pass	Pass	Pass	All 5 participants passed this task without any issue.
		Observation	User did this task very quickly because he was already familiar by then.	User did this task very quickly because he was already familiar by then.	User did this task very quickly because he was already familiar by then.	User did this task very quickly because he was already familiar by then.	User did this task very quickly because he was already familiar by then.	
		Comments	-	-	-	User suggested a "Select all" button or checkboxes to select multiple triples at the same time.	-	
10	Remove the triple (Joe Rogan, genre, Observational comedy).	**Pass/Fail**	Fail	Pass	Fail	Pass	Pass	Only 3 out of 5 participants passed this task. It is probably not very clear how to remove existing triple that was not extracted from a specific article.
		Observation	User took a very long time to figure this out and did not know how to remove an existing triple about Joe Rogan. Even after being given the hint, she did not see the next page buttons.	User initially wanted to use the fact-checker to remove the triple, which could also work. But, he realised that he could also do it from the Entity Explorer.	User was not expecting that he could remove triples from the Entity Explorer page. He was confused on how to do it and he had to be given a hint.	User took some time before going to the correct page, but then doubted himself.	User immediately went to the correct page.	
		Comments	Regarding the page buttons, user argued that she did not see only because of Zoom technicalities.	-	User does not like that DBpedia format needs to be used as an input.	User noticed that the number per page options in the table was not available.	User noticed that the number per page options in the table was not available.	
11	Other than what you have said already, is there anything specific that you want to comment on?		No.	No.	No.	No	User felt that the page name "Add Own Knowledge" is not too descriptive.	-
12	Do you think that the interface is good for users who are trained to use it?		Yes, even as a new user she said this is quite simple, so it will definitely be intuitive for trained users.	Yes, he is confident that everything is intuitive and would be smooth if it was used by a trained user.	Yes, absolutely. He feels that almost everything is self-explanatory.	Yes, he believes that in a professional settings where the user was trained, this would be very beneficial. He said that he did struggle a bit when looking for certain stuff, but if he were to do this again, he would be able to do everything easily.	Yes, it is intuitive enough for him that was not familiar with the web app, so it will certainly be more intuititive for trained users.	The system's interface is intuitive especially for users that have been trained to use it.
13	Is there any other suggestion?		No.	No.	No.	No.	No	-

Appendix F

Requirements Completion Table

#	Requirement	Input	Expected Output	Actual Output	Pass/Fail
			Functional Requirements		
1	The KGU should mirror or update the knowledge graph if the existing open knowledge graph used is updated.	Update on the existing knowledge graph	The locally hosted knowledge graph should be automatically updated.	The locally hosted knowledge graph does not dynamically mirror the existing knowledge graph. It stays static when the mirrored knowledge graph is updated.	Fail
2	The KGU should scrape news articles from trusted news websites periodically, at least every 1 hour if the system is running all the time.	Newly published news articles	The news articles should be scraped periodically and stored in the database.	The news article can be scraped periodically and stored in the database.	Pass
3	The KGU shall extract SPO triples from the trusted news articles and from the user feedback.	Trusted news articles	Triples should be extracted from the news articles and stored in the database.	Triples are extracted from the scraped trusted news articles and stored in the database.	Pass
		User input	Triples should be extracted from the user input.	Triples are extracted from the user input.	
4	The KGU shall be able to add the extracted triples to the knowledge graph if the triples do not exist yet in the knowledge graph.	Triples	Extracted triples should be able to be added to the knowledge graph.	Extracted triples are able to be added to the knowledge graph on user demand. However, if the triples do not have conflicts in the knowledge graph, it can also be added automatically.	Pass
5	The KGU shall be able to modify or remove triples from the knowledge graph if they are conflicting with the extracted triples.	Triples	Triples in the knowledge graph should be modified or removed if found to be conflicting.	Triples can be removed or modified by removing and adding the new triple on user demand if found to be conflicting.	Pass
6	The fact-checker shall extract SPO triples from the to-be-verified news articles.	To-be-verified news articles	Triples should be extracted from the to-be-verified news articles.	Triples are extracted from the to-be-verified news articles.	Pass
7	The fact-checker shall perform fact-checking algorithms on the to-be-verified triples.	To-be-verified triples	The to-be-verified triples should be fact-checked using the fact-checking algorithms.	The to-be-verified triples are fact-chekced using the fact-checking algorithms.	Pass
8	Depending on the fact-checking algorithm, the fact-checker should be able to query the knowledge graph as needed.	To-be-verified triples	The knowledge graph should be queried by the fact-checker when fact-checking the to-be-verified triples.	The knowledge graph are queried by both the exact match and non-exact match fact-checker algorithms when fact-checking the to-be-verified triples.	Pass
9	The fact-checker shall return some form of calculated truthfulness score for the triples based on the fact-checking algorithms.	To-be-verified triples	Some form of truthfulness score of the to-be-verified triples should be returned by the fact-checker.	A category type for every triple that is fact-checked are returned by the fact-checker, where the categories are exact match, possible match, conflicting, and unknown. This could be seen as a stricter version of truthfulness score.	Pass
10	In the fact-checking mode, the UI shall accept a news article in the form of sentences as input.	Sentences	The sentences should be accepted as input by the UI to be fact-checked.	The sentences are accepted as input by the UI to be fact-checked.	Pass
11	In the fact-checking mode, the UI should be able to accept a news article link as input.	Article link	The article link (URL) should be accepted as input by the UI to be fact-checked.	The article link should be accepted as input by the UI to be fact-checked.	Pass

12	In the fact-checking mode, the UI shall send the user input to the Fact-checker component.	User input (sentences/link)	The user input should be sent by the UI to the Fact-checker to be fact-checked.	The user input is sent by the UI to the Fact-checker to be fact-checked, along with the type of algorithm to be used and the extraction scope.	Pass
13	In the fact-checking mode, the UI shall display the truthfulness score of an article received from the Fact-checker.	User input of article	A truthfulness score should be displayed by the UI after receiving it from the Fact-checker.	A category type of the triple is displayed by the UI after receiving it from the Fact-checker.	Pass
14	In the knowledge graph update mode, the UI shall accept triples as input.	Triples	The triples should be accepted by the UI as input in the knowledge graph updating mode.	The triples are accepted by the UI as input in the knowledge graph updating mode.	Pass
15	In the knowledge graph update mode, the UI shall send the input to the KGU component.	User input (triples)	The input should be sent by the UI to the KGU component.	The input is sent by the UI to the KGU component.	Pass
16	In the knowledge graph update mode, the UI should return some form of feedback to the user stating that the knowledge graph has been updated, possibly by showing the related entities.	User input	The UI should display a feedback stating that the knowledge graph has been updated with the user input.	The UI displays a short notification stating that the triples has been added to the knowledge graph. The entity then can be further inspected from the entity explorer view.	Pass
			Non-Functional Requirements		
17	The system shall be easy to use, at least in the fact-checking mode.	-	Users should be able to use the system easily, at least in the fact-checking mode.	Based on the usability testing, the users can use the system easily, especially in the fact-checking mode.	Pass
18	The system shall return the outputs to the user in real time.	-	Outputs should be returned to the user in real time	Outputs are returned to the user quite immediately in a reasonable amount of time.	Pass
19	The system should be able to update the knowledge graph continuously in real time.	-	The knowledge graph should be able to be updated continously in real time.	The knowledge graph is able to be updated continously in real time by scraping news articles periodically.	Pass
20	The KGU and Fact-checker shall be written in Python as it offers plenty Natural Language Processing libraries.	-	Python should be used to develop the KGU and Fact-checker components.	Python is used to develop the KGU and Fact-checker components.	Pass
21	The UI shall be a web interface written in HTML, CSS, and JavaScript.	-	Users should be able to access the interface via web browsers.	Users are able to access the interface via web browsers because the UI is written with React JS.	Pass

Appendix G

Meeting Logs

Meeting 18/08/2020

- Short initial discussion via e-mail where I proposed the idea of the project.
- Suggestions around the interface and how to make the project more feasible were given.

Meeting 05/10/2020

- Discussed initial ideas and overview of the project.
- Gone through the structure of the interim report.
- Discussed an idea where the interface could have 2 modes: fake news checking mode and admin mode where the user can give feedback on whether the fact in the knowledge graph should be updated or not.
- Discussed how the system should be evaluated. It could be by evaluating different methods for identifying facts that I will develop, where one is simple and the other is difficult. The evaluation could be done by asking end-users which method they prefer.
- If human evaluation is needed, I would need to submit an ethical compliance form.
- Talked about how to make the project more feasible, by possibly constraining the relationships that the system is looking for.
- Considered expanding existing knowledge graphs and finding possible news article sources.

Meeting 12/10/2020

- Discussed the project proposal draft that I sent.
- Extended Objective 2: "conduct user testing to evaluate the interface" deemed to be unnecessary as it is somehow a part of Primary Objective 6. Thus, Extended Objective 2 will be removed.
- Initially, Extended Objective 1: "deploy system to cloud server" was also seen as unnecessary. I made a point that if it is a full system, then it needs to be able to run all the time. But it is not a requirement, hence it is an extended objective.
- Discussed how to get the news data, either scrape from news websites or find available datasets.
- Supervisor suggested that I investigate the relationships in existing knowledge graphs and pick some common relationships that also appear in (Covid-19) news.

Meeting 21/10/2020

- Discussed DBpedia and DBpedia Live.
- Talked about entity relationships that can be considered, such as "person"-said-"statement"-at-"date"-at-"place". I should also maybe start with relationships between named entities.
- Supervisor suggested that maybe different newspapers paraphrased statements that made them misleading, which could also be investigated for future works.
- For "A said B" relationships, I should focus on whether it is true that "A said B" or not, but not focusing on whether B itself is correct or not. Then, I could expand on investigating if a paraphrased statement means the same thing as the original statement.
- Discussed two different families of fact-checking algorithms, which are knowledge graph embedding and network flow, and the possibility to compare both.
- Supervisor suggested me to understand the TransE embedding model.
- I was reminded of what the interim report should consist of.

Meeting 26/10/2020

- Explained and discussed the TransE knowledge graph embedding model.

Meeting 2/11/2020

- Discussed details of specific sections needed to be included in the interim report.

Meeting 9/11/2020

- Discussed the feedback for the interim report draft.
- Most of the errors are grammatical or typographical.
- I need to put more on the ethical issues section.
- Clarified the system design, with regards to the Knowledge Graph Updater's interaction with the User Interface and news articles.
- Discussed a bit more on TransE and how it deals with relationships synonymy.

Meeting 23/11/2020

- Discussed about mirroring DBpedia and the possibility of only extracting and loading triples that have specific relations, before loading them into the database.
- Supervisor suggested me to think about whether I want to work via the SPARQL interface or with a lower level/direct implementation to the triples data, because it depends on whether I want to use a lot of functionalities given by SPARQL.
- Spacy can probably be used for named entity recogniser and openie be used for the SPO triples extractor. However, I will probably need to make my own relation extractor, which maps with the type of relations that DBpedia has. Thus, it is important to restrict the relations as a start.
- For next week, find the kind of relationships that often occurs in the news dataset that I am going to use.
- DBpedia Spotlight can also be used for the Named Entity Recogniser, that matches the entities with DBpedia resources. I might also need to combine this with Spacy, in case there is any named entities that is not recognised by DBpedia Spotlight.

Meeting 30/11/2020

- Need to show justifications on why I chose openIE instead of Textacy. Give examples of results of the same texts.
- For the most common relationships (especially "is in" relation) that occur in the dataset, classify their named entities. Also try to match the relationships to the

ones in DBpedia

- If entities/relationships are not available in DBpedia, there is nothing wrong by just adding them. But to start with, try to start with existing ones in DBpedia.
- Noticed that the quality of relationships is not always perfect. Look into some samples and investigate them: MY own extraction vs openIE. Probably worth trying to make the quality better. Maybe Textacy is also good if it returns nothing, compared to openIE that might give noisy data.
- Relations such as "suggested" or "said" tend to be extracted incorrectly.
- For real news articles scraping, Guardian and Times might be good candidates. I need to investigate those websites (and others) for next week, in terms of their structure and freely available content.

Meeting 7/12/2020

- Discussed the feedback for the interim report. There were minor typos (unsupervised vs supervised). There might need to be contingency plan if project not completed.
- The project is ambitious, so need to limit the scope, by limiting the relations.
- Discussed the comparison between Stanford OpenIE, Textacy, IIT OpenIE, and my own extraction.
- To decide which SPO extractor to use: if the extractor doesn't work well on complex sentences, it should at least work really well on simple sentences. For the complex sentences, need to think if it's better to throw nothing or to return noisy extractions. It depends on whether the potential extractions are shown to a user first before updating the knowledge graph or not.
- If using only one system that is consistent (I.e. always extracting the same way, makes the same kind of mistakes), that is probably good enough.
- Many of those relations are time limited. Might want to see relations that are less transient.
- Choose SPO extractor that does the best when extracting relations between 2 (known) Named entities/noun phrases and simple cases first.
- Showed the named entities in "is in" relations. Most of them are places.
- Showed the web scrapers.

Meeting 29/1/2021

- Showed what I have done throughout the Christmas break.

Meeting 5/2/2021

- Discussed what approach I should choose for the fact-checking algorithm: linguistic or logical.
- Discussed a bit on what needs to be included in the poster.
- Supervisor suggested to start developing the UI.

Meeting 12/2/2021

- Showed the current state of the system and the simple UI.

Meeting 19/2/2021

- Discussed what needs to be done in the more complex fact-checking algorithm.

Meeting 26/2/2021

- Discussed the current state of the system and showed the fact-checking algorithms.
- Discussed the need to revamp the UI so that the triples are separated by clear categories.

Meeting 5/3/2021

- Discussed the draft of the poster.
- I need to include higher-level motivation in the poster.
- Some of the parts in the poster may not be needed.
- Discussed evaluation methods, possibly by self fact-checking real news.
- User evaluation may be needed to show that the system does work.

Meeting 12/3/2021

- Showed the current state of the system and the fact-checking algorithm.
- Discussed the structure of the final report and reminded to start writing.

Meeting 26/3/2021

- Discussed on what needs to be included in the evaluation section.
- Further discussed evaluation methodology and to quantify the categories of mistakes.
- Reiterated the importance of introduction and conclusion sections.

Meeting 23/4/2021

- Discussed the feedback on the final report draft.
- I need to break down long sentences, include breakdown in the beginning of each chapter, and include requirement analysis.
- It might be worth to do a user evaluation/usability testing.

Meeting 30/4/2021

- Discussed on what needs to be put in appendices, which are evaluation details, code listing, ethical compliance form, meeting logs, and proposal..
- Discussed the presentation, which needs to include the project challenges and how I overcome them.

Meeting 7/5/2021

- Decided to only include a sample of evaluation to the appendix and the rest into the code submission.
- Reiterated the need to show challenges in presentation.

Appendix H

Proposal

Project Proposal

Using Dynamic Knowledge Graph for Fake News Early Detection

by Albertus Andito

Candidate number: 198910

Supervisor: Julie Weeds

Project Background

News are often about recent events, including events that are still happening. Thus, some knowledge and facts could become invalid in just a short period of time. A fake news detection system should be able to cope with such timeliness of news, where information or facts dynamically change over time. With that, fake news can be recognised early and immediately after the ground truth is updated.

Therefore, this project will involve creating a prototype of a fake news early detection system, which automatically keeps the ground truth updated in real-time inside a knowledge graph. The ground truth will be extracted from trustworthy news articles in the form of SPO (Subject, Predicate, Object) triples.

Another component of this project will be a web application that serves as the interface for the user to check whether a certain statement or news is correct or not, by querying the knowledge graph. The interface could also have a mechanism that allows the user to check whether the ground truth should be updated when the facts are believed to be out-of-date.

The scope of this project should be limited to make it more feasible, such as by limiting the number of predicates or entity types, or perhaps limiting the domain.

Aim

- Develop a fake news detection system where the ground truth is stored in a dynamic knowledge graph and updated in real-time, in attempt to recognise fake news at an early stage.

Primary Objectives

1. Research and evaluate existing knowledge-based fake news detection systems.
2. Identify existing knowledge graphs that are relevant and can be extended for news data.
3. Implement an article extractor which extracts SPO triples from articles and added them to knowledge graph, while also removing outdated triples.

4. Develop and implement at least two algorithms to compare the to-be-verified facts with the knowledge graph.
5. Develop an interface for users to submit articles and check their truthfulness, and to update the knowledge graph.
6. Evaluate the different methods used to automatically verify the news articles.

Extension Objectives

1. Deploy the fake news detection system to a cloud server where it can run all the time.

Relevance

This project is relevant to my degree course, BSc Computer Science (with an industrial placement year) for several reasons. It focuses on research in the field of Natural Language Processing, which I have learnt from the Natural Language Engineering module and will study further in the Advanced Natural Language Engineering module next term. Being able to construct a computer-based system is one of the course learning outcomes. Thus, the Software Engineering module and my industrial placement year experience will also prove to be useful for the development part of this project.

Resources required

- Access to news articles. For testing and development purposes, the number of articles needed is limited. However, to fully achieve Extension Objective 1, unlimited access to news articles might be necessary.
- Cloud resource for running the server to achieve Extension Objective 1.

Bibliography

[1] M. D. Ibrishimova and K. F. Li, “A Machine Learning Approach to Fake News Detection Using Knowledge Verification and Natural Language Processing,” in *Advances in Intelligent Networking and Collaborative Systems*, Cham, 2020, pp. 223–234, doi: 10.1007/978-3-030-29035-1_22.

[2] X. Zhou and R. Zafarani, “A Survey of Fake News: Fundamental Theories, Detection Methods, and Opportunities,” *ACM Comput. Surv.*, vol. 53, no. 5, pp. 1–40, Sep. 2020, doi: 10.1145/3395046.

[3] J. Z. Pan, S. Pavlova, C. Li, N. Li, Y. Li, and J. Liu, “Content Based Fake News Detection Using Knowledge Graphs,” in *The Semantic Web – ISWC 2018*, vol. 11136, D. Vrandečić, K.

Bontcheva, M. C. Suárez-Figueroa, V. Presutti, I. Celino, M. Sabou, L.-A. Kaffee, and E. Simperl, Eds. Cham: Springer International Publishing, 2018, pp. 669–683.

[4] P. Lara-Navarra, H. Falciani, E. A. Sánchez-Pérez, and A. Ferrer-Sapena, "Information Management in Healthcare and Environment: Towards an Automatic System for Fake News Detection," *International Journal of Environmental Research and Public Health*, vol. 17, no. 3, Art. no. 3, Jan. 2020, doi: 10.3390/ijerph17031066.

[5] S. Choudhury *et al.*, "NOUS: Construction and Querying of Dynamic Knowledge Graphs," in *2017 IEEE 33rd International Conference on Data Engineering (ICDE)*, Apr. 2017, pp. 1563–1565, doi: 10.1109/ICDE.2017.228.

[6] S. S. Pavlova, "Using Knowledge Graphs for Fake News Detection," University of Aberdeen, Aberdeen, UK, 2018.

Interim log

Meeting #1 (18-08-2020):
Short initial discussion via e-mail where I proposed the idea of the project. Suggestions around the interface and how to make the project more feasible were given.

Meeting #2 (05-10-2020):
First meeting since the term started. Discussed about the scope of the project, considerations on expanding existing knowledge graphs, how to evaluate the project, and possible news articles sources.