National College of Ireland

Relation Extraction using Using Natural Language Processing and Deep Learning Techniques

MSc Research Project
Data Analytics

Bryan O'Donohoe
Student ID: X20212828

School of Computing
National College of Ireland

Supervisor: Jorge Basilio

National College of Ireland
Project Submission Sheet
School of Computing

National College of Ireland

Student Name:	Bryan O'Donohoe
Student ID:	X20212828
Programme:	Data Analytics
Year:	2022
Module:	MSc Research Project
Supervisor:	Jorge Basilio
Submission Due Date:	15/08/2022
Project Title:	Relation Extraction using Using Natural Language Processing and Deep Learning Techniques
Word Count:	5395
Page Count:	20

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	
Date:	19th September 2022

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Relation Extraction using Using Natural Language Processing and Deep Learning Techniques

Bryan O'Donohoe
X20212828

Abstract

This research study aims to address the issue of relation extraction. The area of relation extraction has various use cases such as question and answering systems, fact checking and the conversion of semi-structured and unstructured text into knowledge bases. There is now more information than ever before available online, and with this comes additional need for machines to translate this data into easily searchable databases. This study describes the approach taken in the end-to-end deployment of a model in which the end user can extract a relation between two named entities on an AWS hosted server. The dataset used to develop this system is taken from the Wikipedia website and contains labelled text for two named entities. The key findings in this report are that a bidirectional encoder representations from transformers model is the optimal solution when trying to extract relations between two named entities from a body of text. There is scope for further research with the development of a hybrid generative and discriminative model, as well as a more optimal deployment of the final application.

1 Introduction

With the advancement of artificial intelligence, machines are ever improving in interpreting the world and converting data points into information in the same way humans do. One area where a human can out perform a machine is in the area of fact extraction. The human ability to infer information about an entity from the full context of text as opposed to just a specific sentence is where machines are at a disadvantage. As it is not possible to know all of the possible types of facts that could be extracted text without a large set of possible combinations, this presents a multi-label classification problem for the machine where the full sample set of possible relations is unknown. The challenge is to generate possible relations based on the context of the sentence of paragraph by either directly extracting the relation from the body of text or else deriving it using a synonym.

1.1 Motivation and Project Background

With more text information than ever before online, there is an ever increasing corpus in which data mining and machine learning models for fact extraction can be trained and tested. To date, many of the solutions proposed have utilised a dependency tree method in which the the grammatical and structural properties of a sentence is analysed to identify related words and describe the relationship between them. As with most natural language processing tasks, some form of deep learning is used in most proposed solutions. The target variable in a task of this

nature indicates that this is similar to multilabel classification in which this full list of known labels is not known. This has meant that some of the more successful research studies have utilised a combination of a generative and discriminative models to extract facts from the body of text. This neural network design effectively infers facts for a given entity and scores them to yield the optimum results. The ability to generate labels, extract context and ignore irrelevant information are inherently human traits, however the gap between the performance of machines and humans is narrowing. This study aims to identify processing, model and evaluation changes to state of the art solutions in this research area.

1.2 Research Question

Much of the existing research utilises the Semeval 2010 dataset in which two named entities are label in a body of text and the relation between them is extracted. The research will attempt to build on existing efforts to answer the following 3 research questions:

- To what extent can facts about a named entity be extracted from a body of text?
- Is a convolutional neural network or a idirectional encoder representations from transformers model more effective in extracting facts from a body of text?
- Can a model be used in combination with named entity recognition to extract facts from a corpus of text?

This study will provide an in-depth analysis of the state of the art studies and propose a solution that can be deployed in an API to extract facts from a body of text.

1.3 Research Objectives

Table 1: Objectives and techniques for objective evaluation

obj	**Brief Description**	**Techniques and Evaluation Metric**
1	Literature review of recent studies in the area of fact extraction	N/A
2	Clean and pre-process the dataset	N/A
3	Encode the input and target data	N/A
4	Computation of models to extract factual information from text	F1-score
5	Load model on to AWS Server	F1-score
6	Deploy model into API	F1-score

This study is based on the wikifact dataset provided by the Google research team. This analysis is limited as the data used is wholly obtained from Wikipedia and the writing style may not generalise to text using an alternative style guide. The contribution of this study is the application of a bidirectional encoder representations from transformers model to correctly label a relation between two named entities on an Amazon Web Services hosted web server and to analyse it's improvement on the existing methods. This report will critically evaluate the latest literature in this field, followed by a detailed description of the methodology used in the implementation of the solution, the results will be presented and evaluated and finally there will be a discussion and conclusion of the findings.

2 Related Work

Fact and relation extraction have been areas of research for many years now, however with the normalisation of the internet, there is now more information than ever available online in unstructured format. This literature review looks at the main studies in the area broken down by the motivation to carry out research in this area, the pre-processing techniques utilised and the various modelling techniques that have been used in answering this question in the past.

2.1 Motivation for Studying Fact Extraction

This subsection outlines the main motivations for studying relation and fact extraction in the field of Natural Language Processing as they pertain to: fact checking, search engines and question and answering systems.

The demand to fact check claims made online has increased in recent times due to the increased volume of information and the simplicity at which it can be shared. Thorne et al. (2018) stated that in scientific publications, product reviews, journalism and question and answering systems, that the ability to verify statements made is important moving forward. This research states that developing a model to highlight information that is factually incorrect is difficult. The proponents of this studied mining and developed a scalable dataset which can be utilised to train train different components of fact extraction, which when used in tandem with a sufficiently large knowledge database can be used to check for factual inaccuracies of either journalistic or social media articles or posts. Most datasets developed to assist in building models for fact extraction require a large human input to label the datasets. The human input for this and other datasets pose the problem of being time consuming and prone to human error.

With more and more use of social media, socio-political deception, fake news and online rumours have increasingly become an area of concern (Nie, Chen and Bansal 2018). Recently, a court case, in which Facebook were the defendant, was brought to the courts for a lack of monitoring of false allegations about a high profile media personality to be posted on the platform. This has highlighted the responsibility for social media platforms to take ownership and monitor their site for potentially damaging content posted[1]. Jiang (2020) attempted to create an automatic fact checking platform which could be used on social media. For this case, has the platform been able to correctly highlight the campaign for containing factually incorrect statements, then the lengthy legal process and subsequent costs of said trial could have been negated.

A second area that relies heavily of the ability to extract factual information and relations from large structured, semi-structured and unstructured sources are search engines. Open information extraction, which is a method most commonly used in search engines, relies on heavily supervised knowledge bases (Sawant and Chakrabarti 2013). The knowledge bases in question here mainly assume facts can be identified with verb-based phrases and consequently will perform poorly when a noun-based sentence contains the fact or relation.

A methodology used in an attempt to improve performance for noun based relations is proposed by Yahya et al. (2014). The study here shows a new methodology which aims to improve performance in this area relating to nominal attributes. Distant supervision was used in this paper with the aim to create a solution which could better generalise to as yet unseen facts. The downfall of utilising distant supervision is the need for such a large volume dataset to form these as yet unseen relations.

[1] Independent.ie 2022.

The majority of question and answering systems analysed in this study tend to create a list of ranked documents in which the information is contained, this is known as document retrieval. A corpus (large database of structured text) based solution by Jijkoun, Mur and Rijke (2004) looks to move towards information retrieval as opposed document retrieval. This study has the correct goal in information vs document retrieval, however it utilises mainly inferential or shallow methodologies which while not costly to use, do not provide the outputs with the same level of accuracy as deep learning models, which will be shown below.

This sections has illustrated the uses for fact and relation extraction across the domains of search engines, fact checking and question and answering systems, with much of the advancements in the area being contributed in the last 10 years.

2.2 Pre-processing Techniques in Relation and Fact Extraction

This section analyses at the various methodologies utilised in the transformation of text before relation extraction is carried out by various modelling techniques.

Much of the research carried out in the area of relation extraction use a dependency tree method where the grammatical and structural makeup of a sentence is analysed in order to find the path of linking word words in the sentence and to use this additional information to describe the type of relation between them. As shown below in Figure 1, a basic dependency tree is shown in which it can be seen that the presence of the black will change the meaning of the word car if it prior to that word in the sentence.

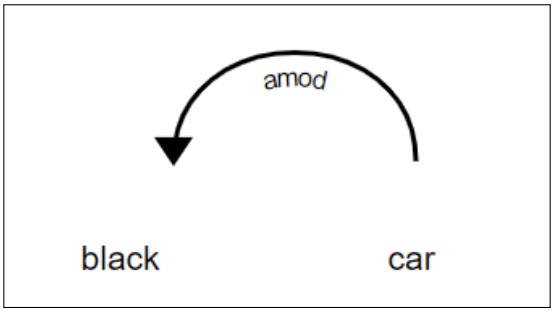

Figure 1: Dependency tree representation of the link between two words, from Towards Data Science, Shivanee Jaiswal 2022

A study that relies on a dependency tree method was carried out by Gamallo, Garcia and Fernández-Lanza (2012). The study used chunking of phrases and part of speech tagging (POS) to create supplementary features in the data transformation process. This implementation relies upon unsupervised learning, open information extraction and was carried out across multiple languages. While the number of languages analysed here is impressive, resources could have been better utilised by analysing just a single language as the structure and grammatical differences that pertain to different languages could be misrepresented in the transformations carried out during this study. Dependency tree methods that were used here can lend itself to other issues as words that are unrelated to the named entities may introduce additional noise which could affect the modelling process.

Various studies have attempted to tackle the problem of noise in the data through the use of careful pruning. K. Xu et al. (2015) proposed a methodology of taking the smallest dependency path that links the named entities in a statement. This attempts to remove redundant words from the sentence. As Y. Xu et al. (2015) pointed out in it's research, unless there is careful pruning

undertaken, there is a chance that critical information from the data could be inadvertently removed during the transformation stage.

Another option opposed to the use of dependency trees was tested by Ding (2020) in which they proposed a transformation which was dependency free. This looks to cut down the time taken to process. This method places emphasis on information aggregation and filtering in the data prior to modelling. The system made an attempt at the removal of what would be thought of as insignificant information from the sentence and therefore reducing the strings to basic information. This analysis achieved similar performance as dependency tree models, however it neglects to analyse the cases where the results were different from a dependency model solution, something which is a shortcoming of the researchers. It could be suggested that an aggregation and filtering method used in this study has masked faint elements in the text that could be useful in identifying factual information.

This section has shown that the pre-processing techniques used in relation extraction largely utilises some form of dependency tree methods, however there are alternative methods in the research which could be further investigated.

2.3 Modelling Techniques in Relation and Fact Extraction

This section analyses at the various modelling techniques used in relation and fact extraction. This section analyses the model types, schema and evaluation methods utilised in this area of research.

Distant supervision is the most common technique used in relation extraction research. This involves the knowledge database Freebase, a Google library, which is used as a weakly labelled training dataset. Using this method of learning allows for yet unseen relations between two named entities in a specific body of text, and as such will allow for better generalisation. The model proposed by Surdeanu et al. (2012) uses this in a model for multi label relations as they suggest that the model will learn the co-dependency of some labels that are commonly generated together. This method unfortunately has a major downfall in that it can create unsymmetrical links between relations, e.g. president-of implies citizen-of but not vice versa. This problem with unsymmetrical relations was used as the starting point for research by Riedel et al. (2013) where a universal matrix schema was designed. This proposed an alternative method of solving the problem of substandard labelling or unlabelled data by combining the complete input data into a single matric format. This research achieved a comparable improvement of 10% to state of the art studies using distant supervision result formulated by Surdeanu et al. (2012) in precision. The examples analysed here all are subjected to the problem of using poorly labeled datasets or unsupervised at all, as any method of distant supervision will give weak positive examples, and a large amount of false negatives.

There is much debate over whether a discriminative, generative or a hybrid model should be used in relation extraction, Ajao, Bhowmik and Zargari (2018) argued that a combined approach of a recurrent neural and convolutional network was the optimum solution for fake news detection on Twitter. A comprehensive survey was carried out by Islam (2020) in which the various models utilised across five main categories; spam, false information, fake news, rumor and disinformation. This survey notes that in the use case of fake news detection, a hybrid model was mainly used by other researchers. This survey provides a useful starting point for specific models, depending on the suitability of the data found by researchers as it can allow researchers to quickly get an understanding of the model makeups for various use cases. Neural network models are definitely the state of the art technology and most commonly used in this and other types of NLP tasks, the black box nature of them can lend itself to unexplainable

results and it is a cause for concern in this area as this can be a challenge when presenting solutions to a wider business function.

Hu (2011) attempted a unique method for this problem as a prior likelihood was created for a relation between two named entities in which it utilised Bayesian statistics. This is in contrast to deep learning approaches in that they use an inference classifier as opposed to a linear classifier. A good performance on multi label entity relations was achieved in this study and as such an inference layer could be used in conjunction with deep learning models to yield a better result than either approach as a stand alone method.

During the translation of model outputs back into human readable facts, deep learning models often struggle by creating artificial relations. Santos (2015) proposal of creating pairwise rankings of the predicted entity relations and in the process ignoring non human readable ones and only selecting the highest ranked relation that was also a viable one. This when used in tandem with an inference method similar to the one outlined by Hu (2011) could produce a simple and facts model without compromising performance in this area of research.

Goodrich (2021) proposed a model based method for the evaluation of relation extraction. This evaluated itself against the state of the art metrics such as Recall-Oriented Understudy for Gisting Evaluation(ROUGE) and Bilingual Evaluation Understudy (BLEU) by a manual human comparison. Proposed methodologies are all susceptible to a substandard performance given an alternative writing style. A proposed solution that can better generalise to as yet unseen writing styles could be an area to analyse for future studies in this domain.

This section has shown that the modelling techniques carried out in the areas of relation and fact extraction mainly use neural networks, however there is perhaps a possibility to explore a hybrid of inferential models to create a more explainable final model, that ultimately can be more widely be accepted in a business use case.

3 Methodology

This research implements the methodology of Cross-Industry Standard Process for Data Mining (CRISP-DM). CRISP-DM allows the flexibility to move between the different stages, while also providing a solid framework to follow for data mining. The 6 stages of CRISP-DM are outlined below in Figure 2.

3.1 Problem Understanding

Fact extraction is key in turning unstructured and semi-structured data into structured information. Being able to extract factual information about entities allows one to build a structured database of searchable information, be it for facts or relationships between entities. As described in the literature, there is entity linking required in order to describe which tokens in a string of text pertain to a given entity, e.g. "the prime minister", "Boris" or "Johnson" may all relate to "Boris Johnson" so it is important that if a relation between one of these words is detected that this is reflected in the modelling approach.

3.2 Data Understanding

The dataset obtained from the OpenNRE research team contains 56,000 records of inputs and targets. The dataset is comprised of sentences ranging between 5 and 46 words in length. Each data entry is of JSON format and contains an input sentence, the location of two entities in the

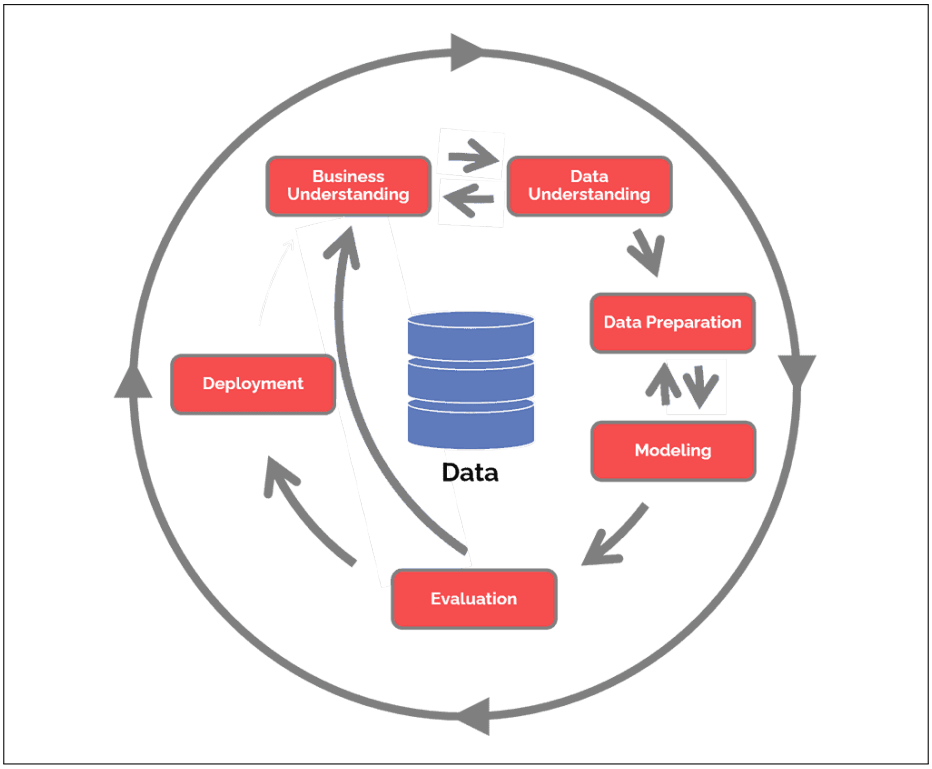

Figure 2: CRISP-DM metholody flow chart, from Data Science Process Alliance 2022

string and a label for the relationship between the pair. The label is one of 80 possible relations. The input is detailed below in Fig. 3.

```
"token": [ "XYZ",
           "was",
           "married",
           "to",
           "ABC",
           "." ],

"h": { "name": "XYZ",
       "id":   "Q0001",
       "pos": [ 0,
                1 ] },

"t": { "name": "ABC",
       "id":   "Q0002",
       "pos": [ 4,
                5 ] },

"relation": "spouse"
```

Figure 3: JSON layout of data

3.3 Data Preparation

As shown in the literature review, the data preparation is an important stage in the accurate extraction of facts from a body of text (Gamallo, Garcia and Fernández-Lanza 2012). The data will be cleaned, transformed and visualised using various Python libraries such as Pandas, Numpy, NLTK and Spacy. Tensorflow and Keras will be used to build deep learning models

as they are sufficiently sophisticated and user-friendly for this use case. The data will be data mined using a Python notebook as this allows interaction and readily visible results.

3.4 Modelling

As with most natural language processing tasks, deep learning models will be primarily used in the development of a solution to this problem. As the literature has shown it will require a discriminative model to solve this problem (Islam 2020). The discriminative models likely to prove useful are bidirectional encoder representations from transformers (BERT) and convolutional neural networks (CNN).

3.5 Evaluation

The predictive performance of the chosen model will be evaluated on unseen validation data when building the models, and the holdout test dataset once the final deployment is completed. As this is a multilabel classification problem, the optimum model will be chosen using a combination of precision, recall and F1 score, as accuracy alone cannot be used to determine the optimum model (Goodrich 2021). Precision, recall and F1 are given by the below equations:

$$precision = \frac{True positive}{True positive + False positive}$$

$$recall = \frac{True positive}{True positive + False negative}$$

$$F1 = \frac{2 * precision * recall}{precision + recall}$$

3.6 Deployment

The final model is to be deployed using an Amazon Web Services EC2 (Elastic Compute) engine. This will allow the fact extraction model to return the relation between two named entities with a corresponding confidence level.

4 Design Specification

Figure 4 below outlines the design flow of this research study with a visual representation of the various components.

The 7 steps in the design of this solution are:

- Data extraction - obtaining relevant data to answer the question of fact extraction
- Data cleaning - standardisation of data to remove noise
- Data pre-processing - any techniques such as tokenization, pre embedding or transformations that were required to feed into the model stages
- Exploratory data analysis - analysis of the data to understand the shape, distribution and contents of the data. This will re inform the cleaning and pre-processing required
- Model Building - building and tuning the various models to test
- Validation - testing the models on the holdout validation data set
- Deployment - deploying the final model into an AWS hosted application

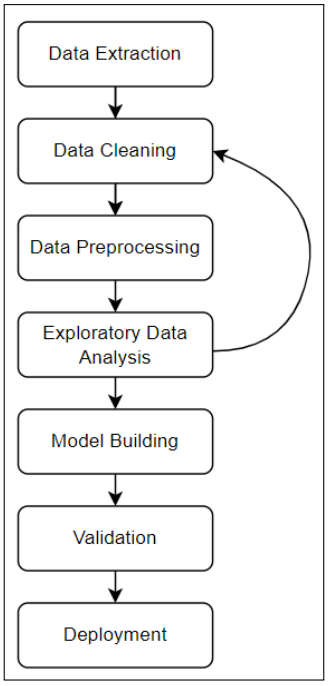

Figure 4: JSON layout of data

5 Implementation

5.1 Introduction

This section shows the research setup, pre-processing, data exploration, how the predictive models have been implemented, evaluated and deployed, and the results of this evaluation.

5.2 Hardware and Set-up

As discussed in chapter 3.2, the data in question is given in the form of a dictionary format within a text file. This was downloaded and loaded into a notebook for analysis. Python was chosen as the programming language of choice. For this analysis, the OpenNRE package (Han et al. 2019) was utilised to streamline the data manipulation and model building processes.

5.3 Data Pre-processing

As discussed in chapter 3.4, the type of neural networks evaluated for this research will be discriminative models. As such, this means that the input entries for the dataset will need to translate the data such that it can be read by both a discriminative model. The data contains the labelled entities, the position within the list of tokens in the sentence where the entities are location and the labelled relation between the two entities.

Parts of speech (POS) tags were added to the raw text in order to build up dependency graphs (Gamallo, Garcia and Fernández-Lanza 2012). As with all natural language processing tasks, encoding was also carried out to read the data into the models. The method used in this research tokenizes the words into pieces of its words. This uses a greedy longest-match-first algorithm to perform tokenization using the given vocabulary (Han et al. 2019). This means that the input string will be returned as pieces of word tokens as shown below in Table 2

Pre tokenization	Post Tokenization
"Sean went to the beach"	"Sean", "went", "to", "the", "beach"

Table 2: Pre and post tokenization

In order to analyse the n-grams in the dataset, a list of stop words was provided such that the exploration could be carried out both including and excluding the stop words.

5.4 Data Exploration

On inspection of the data, it is not surprising to see that there is a large link between mentions of specific key words present in the string of text and the labelled relation if this is analysed on a one-way basis. For example, an analysis of the 630 data points in which the relation between the two entities was labelled as a "child", figure 5 shows that occurrences of keywords would yield quite a good result without the use of any more sophisticated modelling techniques. Further examining of this elementary flag yields only a 0.24 correlation between a mention of one of the key words and the "child" label which suggests that something more sophisticated will need to be done.

	Keyword								
Label	son	daughter	father	mother	parent	dad	mom	child	Total
Child	317	180	94	66	9	3	1	50	630
Percentage	50%	29%	15%	10%	1%	0%	0%	8%	100%

Figure 5: Word count for keywords in statements in which the relation extracted was "child"

Through the use of the Displacy library within the Spacy package (Honnibal and Montani 2017), dependency graphs can be built to visualise the structural makeup of various sentences. These graphs allow the user to see how various parts of speech tags interact with each other and how the named entities are linked. As shown below in Figure 6, it is shown that the two named entities, Sean and Johnathan, are linked by an auxiliary token, a noun and a preposition. The noun being the relationship between the two named entities, with the auxiliary token and the preposition describing the direction of the relationship.

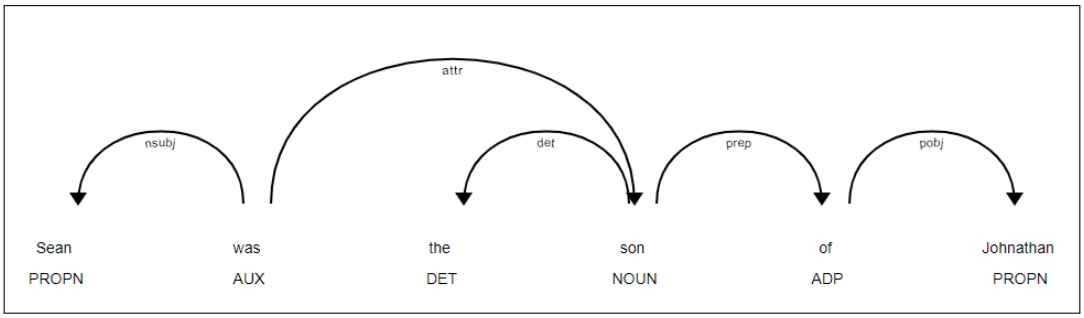

Figure 6: Dependency graph for string representation of relationship between named entity "Sean" and named entity "Johnathan"

A word density plot was generated for the dataset with the character distribution of the number of characters in each sentence analysed here. This plot is outlined below in figure 7.

As shown below the distribution is approaching a normal distribution with a mean value of 132 characters per string of text to analyse.

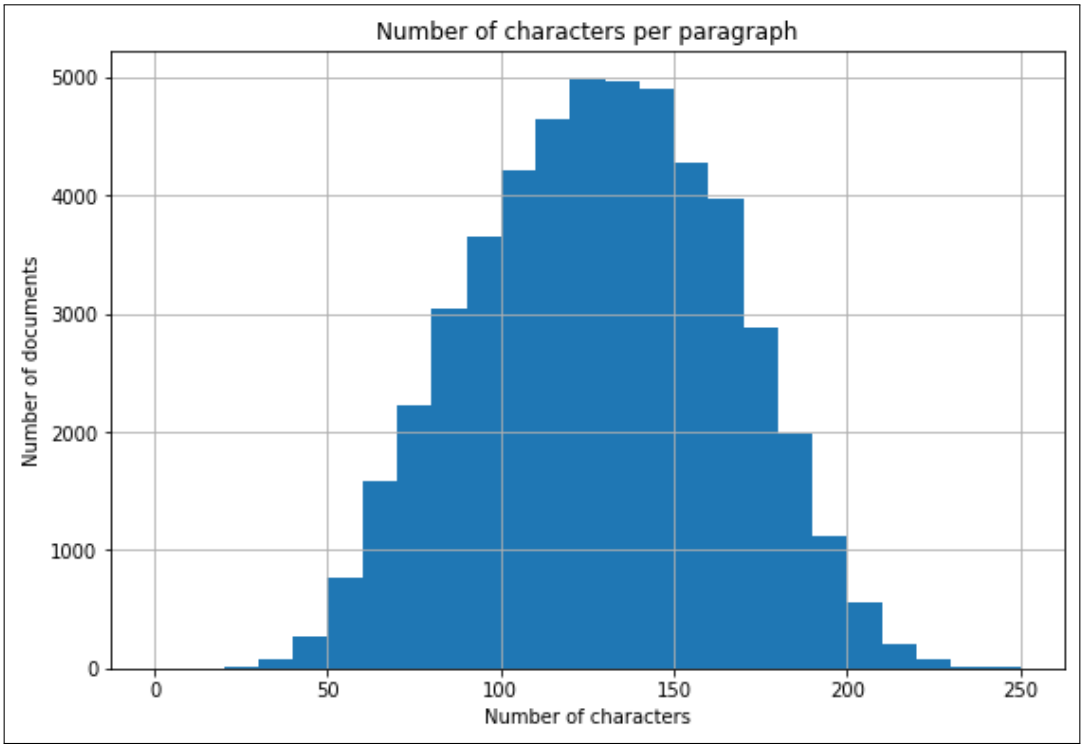

Figure 7: Word density plot

As with most natural language processing tasks, analysis of n-grams was carried out next. Figure 8 below illustrates the top 1-gram in the dataset with no removals of words. It is no surprise to see that stop words are the most common occurring in the dataset being utilised here.

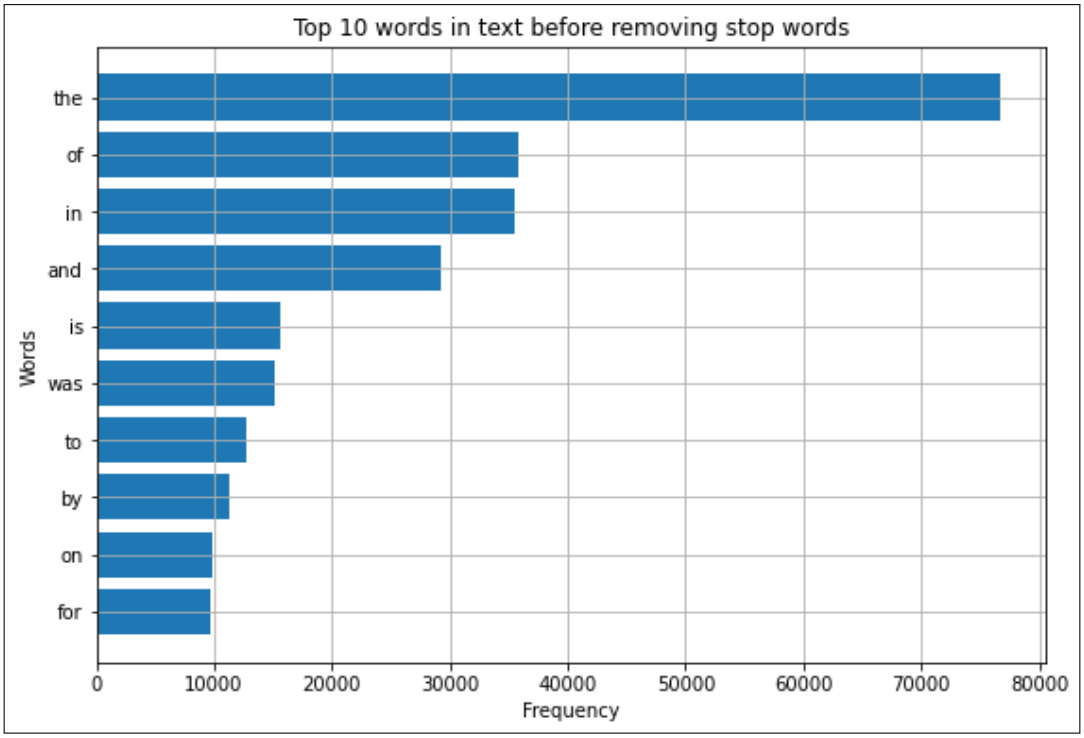

Figure 8: Top 1-gram including stopwords

Figure 9 below illustrates the 1-gram distribution with the stop words removed from the dataset.

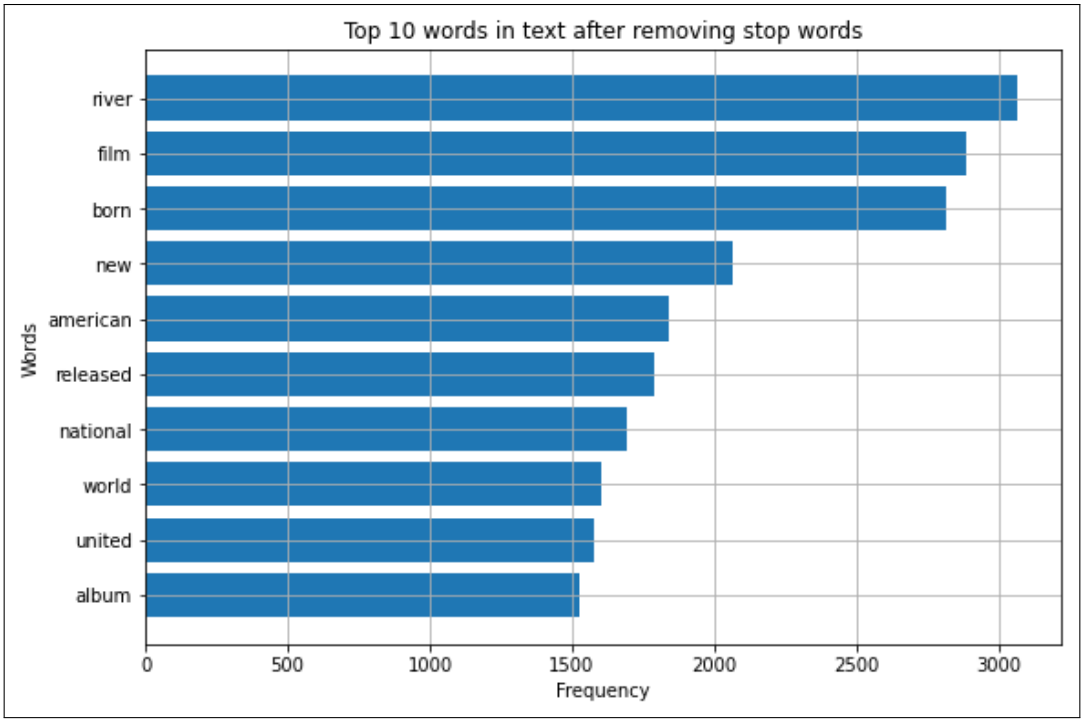

Figure 9: Top 1-gram excluding stopwords

Figure 10 below illustrates the top bi-gram in the dataset with no removals of words. It is again no surprise to see that stop words are the most common occurring in the dataset being utilised here, with "united states" being the outlier here.

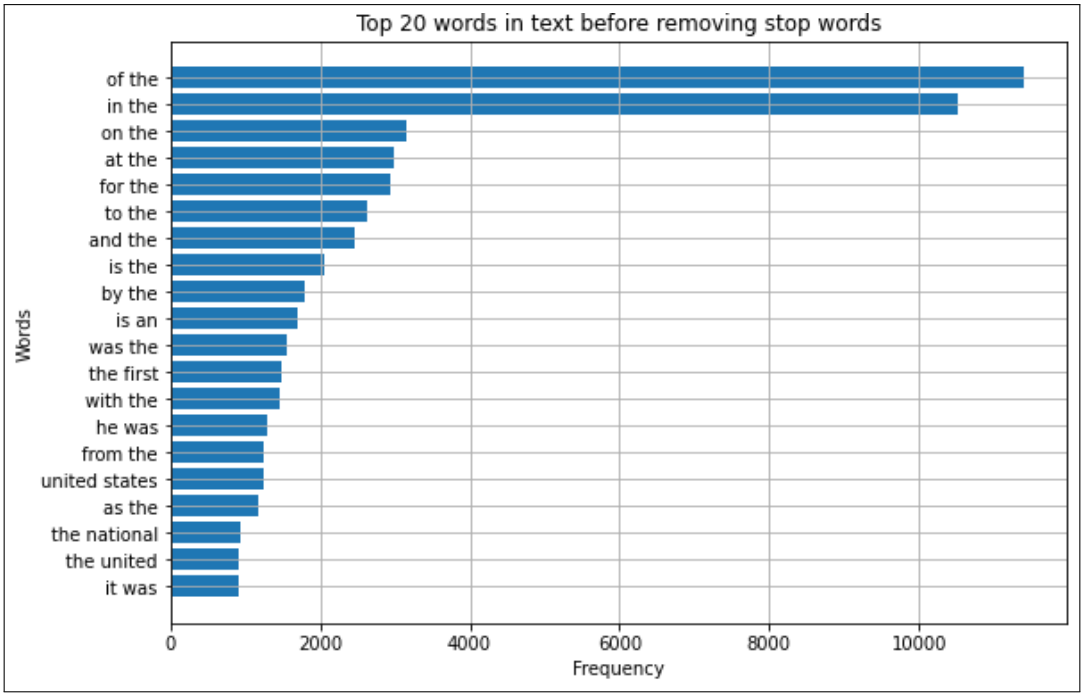

Figure 10: Top bi-grams including stopwords

Figure 11 below illustrates the bi-gram distribution with the stop words removed from the dataset. It is shown here that two locations are the top two occurences in the dataset which may mean a potential overfitting on location facts in the modelling process which is something that may need to be checked during the validation stage.

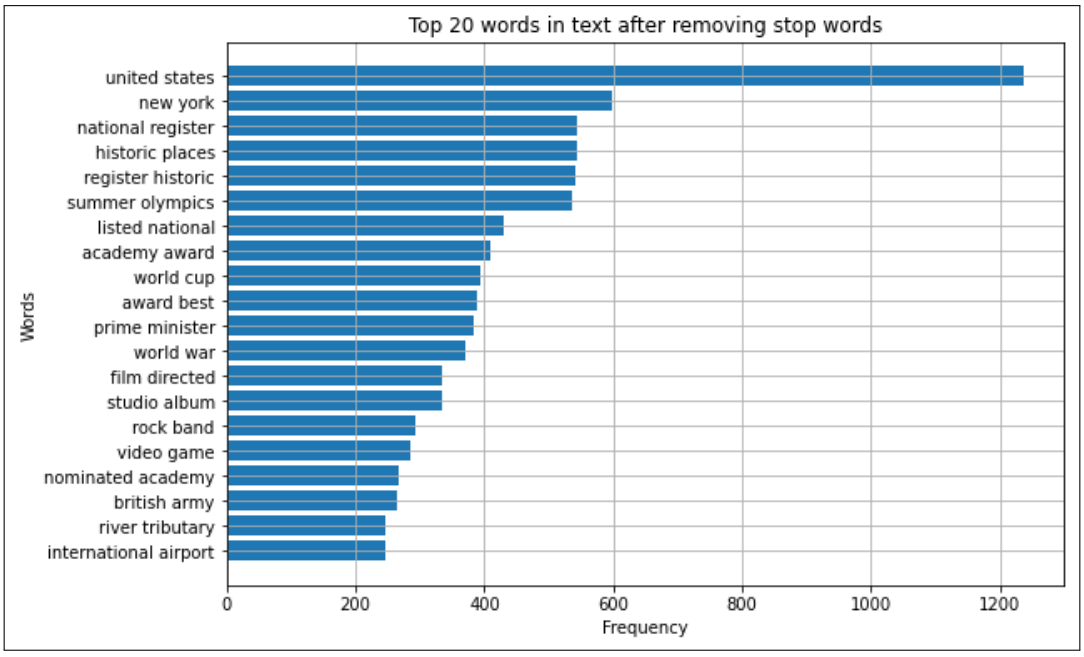

Figure 11: Top bi-grams excluding stopwords

As with the word density, one way analysis and the n-gram plots, this information provides additional granularity which can be used in the validation process to check if there is an under or over fitting on certain characteristics that may be unique to this dataset as ultimately the goal is to provide a generalised useful model.

5.5 Implementation, Results and Evaluation of Benchmark Model

This section outlines the implementation, evaluation and results of the benchmark model used in this study.

5.5.1 Implementation

In order to get a baseline to compare and contrast the experiments, a baseline model was utilised to obtain values for precision, recall and F1 score. As part of the OpenNRE package, a pretrained model was provided with the package. This model was loaded into a notebook and applied to the validation holdout set. The predicted values were downloaded into an excel file where pivots were generated for the different labels and metrics.

5.5.2 Results

The results of the benchmark model are outlined in the following section. Figures 12 and 13 shown below illustrate the top and bottom 10 performing categories in the results of the benchmark model in which the overall F1 score was 0.74.

Top 10 results	F1 score
heritage designation	1
constellation	0.992701
crosses	0.992126
position played on team / speciality	0.991304
taxon rank	0.978417
competition class	0.976
voice type	0.971963
nominated for	0.970588
military rank	0.964286
successful candidate	0.953846

Figure 12: Top 10 performing F1 scores on benchmark model

Bottom 10 results	F1 score
said to be the same as	0.503817
follows	0.5
spouse	0.493274
screenwriter	0.48366
sibling	0.444444
father	0.430233
has part	0.420168
child	0.380488
owned by	0.318182
part of	0.31405

Figure 13: Bottom 10 performing F1 scores on benchmark model

5.5.3 Evaluation

As detailed in the previous section the overall micro-F1 score for the benchmark model is 0.74. It can be seen in figure 12 that a perfect score was obtained on the "heritage description" label, however there are some poor performing labels with 8 labels achieving a score below 0.5 as shown in figure 13.

5.6 Implementation, Results and Evaluation of CNN Model

This section outlines the implementation, evaluation and results of the CNN model used in this study.

5.6.1 Implementation

As outlined in the methodology, a CNN model is one of the more common models used in deep learning as it is less resource consuming to implement. The model utilised in this experiment contained the following configuration as outlined below in Table 4

Parameters	Values
Encoder	word2vec
Padding	1
Kernel Size	3
Dropout	0.5
Hidden layer size	230
Max length size	128
Word embedding size	50
Layers	1D convolutional pool and max pooling

Table 3: CNN setup configuration

5.6.2 Results

The results of the CNN model are outlined in the following section. Figures 14 and 15 shown below illustrate the top and bottom 10 performing categories in the results of the CNN model in which the overall F1 score was 0.69.

Top 10 results	F1 score
constellation	0.992593
position played on team / speciality	0.991304
voice type	0.981132
heritage designation	0.977099
competition class	0.96063
taxon rank	0.956522
nominated for	0.955882
instrument	0.952381
crosses	0.939394
military rank	0.932127

Figure 14: Top 10 performing F1 scores on CNN model

Bottom 10 results	F1 score
instance of	0.421053
headquarters location	0.410256
location	0.40678
father	0.381503
has part	0.368421
sibling	0.338308
spouse	0.244186
child	0.207317
owned by	0.206897
part of	0.153846

Figure 15: Bottom 10 performing F1 scores on CNN model

5.6.3 Evaluation

As detailed in figure 14, there are a number of quite high scoring categories with the CNN model. There are however 16 labels of the 58 that contain an F1 score of less than 0.5.

5.7 Implementation, Results and Evaluation of BERT Model

This section outlines the implementation, evaluation and results of the BERT model used in this study.

5.7.1 Implementation

As outlined in the methodology, a BERT model is more likely to achieve optimal results for this task at the cost of a more resource demanding model building process. The model utilised in this experiment contained the following configuration as outlined below in Table 4

Parameters	Values
Learning rate	0.2
Max epochs	3
Padding	1
Kernel Size	3
Dropout	0.5
Hidden layer size	768
Max length size	128
Word embedding size	50

Table 4: CNN setup configuration

5.7.2 Results

The results of the BERT model are outlined in the following section. Figures 16 and 17 shown below illustrate the top and bottom 10 performing categories in the results of the BERT model in which the overall F1 score was 0.86.

Top 10 results	F1 score
heritage designation	1
position played on team / speciality	1
taxon rank	1
competition class	1
constellation	0.992701
nominated for	0.992481
crosses	0.992248
voice type	0.990476
member of political party	0.983607
head of government	0.981481

Figure 16: Top 10 performing F1 scores on BERT model

Bottom 10 results	F1 score
operator	0.719298
location	0.714286
country	0.704
follows	0.703125
sibling	0.676471
work location	0.661157
screenwriter	0.607843
owned by	0.588235
residence	0.571429
part of	0.504065

Figure 17: Bottom 10 performing F1 scores on BERT model

5.7.3 Evaluation

As show in figure 16, the BERT model yielded some perfect F1 scores of 1 in 4 of the 58 potential categories. In contrast to the benchmark and CNN models, figure 17, not a single category yielded a score below 0.5.

6 Deployment

The deployment of the model into a web based user interface utilised the Streamlit package and an AWS EC2 engine which is outlined in this section.

The EC2 engine used to host the application remotely is outlined below in Table 5:

The application was coded using the streamlit package in which a .py script was created using various streamlit functions in the UI which accept arguments, print warnings and print the response of the model when called upon.

The final application is shown below in figure 18. The application returns the predicted relation between the two named entities entered by the user with the confidence level returned as a percentage.

Configuration Name	Setting selected
Answer private resource DNS name	IPv4
Instance type	t2.micro
Platform	Ubuntu
Platform details	Linux
Memory	16GB

Table 5: AWS configuration settings

Figure 18: Final app deployed

7 Discussion

As shown in figures 12, 13, 14, 15, 16, 17 above, there is some commonality between the top performers in the various models as expected which shows that some of the classes are easily identifiable. Heritage designation, position played on a team and nominated for all have high yielding scores across the models which as even to the human eye they are unique relationships in their own right. Similarly, the models struggled to identify some of similar relations such as father, sibling, spouse and child. These relationships would have similar characteristics in the structure of a sentence with mentions of familial ties being common across all 4 of the labels.

As the literature has shown and backed up by this study, a bidirectional model will perform very well in a natural language task. This is owed to the the bidirectional makeup of the English language where words in a sentence can effect previous and future words to come. This creates a deeper sense of the structure of the sentence for the model and aims to mimic the contextual reading of a sentence that comes naturally to humans.

8 Conclusions and Future Work

In conclusion, one can say that a bidirectional encoder representations from transformers (BERT) model is at the forefront when it comes to correctly labelling the relationship between two entities in a given string of text. The difference between the convolutional (CNN) model and

the BERT model was substantial with a difference of 0.1 in the micro-F1 score. There was a trade off between the CNN and the BERT models in that the BERT model took 1.5 hours to fit whereas the CNN model could be fitted in less than 15 minutes. This meant that optimization of the BERT parameters could not be carried out to a satisfactory standard as there was such a large time cost involved in each iteration of the model. It can be said that if a user would prefer a faster model building process that they could utilise a CNN model however a poorer performance would be yielded.

The research objectives of this study were handled as below:

- Literature review - The most relevant literature in the past 10 years was critically evaluated in section 2
- Clean and pre-process data - standardisation of data to remove noise was carried out prior to implementation
- Encode variables - Stages were carried out prior to the model building stage
- Model building - Both the CNN and BERT models were build and had their scores compared to the benchmark model
- Load model to AWS - Model loaded into an AWS hosted EC2 engine
- Deploy into application - A streamlit app was deployed for access remotely

While this research was efficient in the supervised learning task, the relation was not one of the 58 labels that was contained in the training dataset, the model would incorrectly select the nearest relation. In order to achieve the optimal generalisation for fact relation extraction, the model would need to be a hybrid model in which it is both generative (to generate the unseen labels) and discriminative (to score the labels on the probability)

The application in which the model was deployed in also contained performance issues. It can take up to 30 seconds for the application to calculate the relation between the two named entities entered by the user which if this was to be utilised to generate a knowledge base from unstructured data would lead to a large time cost.

8.1 Future Work

Future analysis in this area should look to create a hybrid discriminative and generative model which will lead to better generalisations on as yet unseen data. More thorough optimization of the parameters could also lead to a higher F1-score for the BERT model. A more optimized method of calling the model in the application could lead to a better performance for the final application.

References

Ajao, Oluwaseun, Deepayan Bhowmik and Shahrzad Zargari (2018). 'Fake News Identification on Twitter with Hybrid CNN and RNN Models'. In: *Proceedings of the 9th International Conference on Social Media and Society*. URL: https://arxiv.org/ftp/arxiv/papers/1806/1806.11316.pdf.

Data Science Process Alliance (2022). *CRISP-DM Diagram*. [Online; accessed June 3rd, 2022]. URL: https://www.datascience-pm.com/crisp-dm-2/.

Ding, Lifang (Dec. 2020). 'FAT-RE: A faster dependency-free model for relation extraction'. In: *Journal of Web Semantics*. URL: https://www.sciencedirect.com/science/article/pii/S1570826820300366.

Gamallo, Pablo, Marcos Garcia and Santiago Fernández-Lanza (Apr. 2012). 'Dependency-Based Open Information Extraction'. In: *Proceedings of the Joint Workshop on Unsupervised and Semi-Supervised Learning in NLP*. Avignon, France: Association for Computational Linguistics, pp. 10–18. URL: https://aclanthology.org/W12-0702.

Goodrich, Ben (May 2021). 'Assessing The Factual Accuracy of Generated Text'. In: URL: https://arxiv.org/abs/1905.13322.

Han, Xu et al. (2019). 'OpenNRE: An Open and Extensible Toolkit for Neural Relation Extraction'. In: *Proceedings of EMNLP-IJCNLP: System Demonstrations*, pp. 169–174. DOI: 10.18653/v1/D19-3029. URL: https://www.aclweb.org/anthology/D19-3029.

Honnibal, Matthew and Ines Montani (2017). 'spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing'. To appear.

Hu, Wenfei (Nov. 2011). 'NLIRE: A Natural Language Inference method for Relation Extraction'. In: *Journal of Web Semantics*. URL: https://www.sciencedirect.com/science/article/pii/S1570826821000561.

Independent.ie (2022). *'Ball is in Facebook's court now', says solicitor after Miriam O'Callaghan receives 'unreserved' apology over bogus skincare ads*. URL: independent.ie/irish-news/courts/ball-is-in-facebooks-court-now-says-solicitor-after-miriam-ocallaghan-receives-unreserved-apology-over-bogus-skincare-ads-41387079.html (visited on 26/02/2022).

Islam, Md Rafiqul (Oct. 2020). 'Deep learning for misinformation detection on online social networks: a survey and new perspectives'. In: *Social network analysis and mining*. URL: ncbi.nlm.nih.gov/pmc/articles/PMC7524036/.

Jiang, Shan (Apr. 2020). 'Factoring Fact-Checks: Structured Information Extraction from Fact-Checking Articles'. In: *Proceedings of The Web Conference 2020, Taipei, Taiwan, April 2020*. URL: https://dl.acm.org/doi/fullHtml/10.1145/3366423.3380231.

Jijkoun, Valentin, Jori Mur and Maarten de Rijke (Aug. 2004). 'Information Extraction for Question Answering: Improving Recall Through Syntactic Patterns'. In: *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*. Geneva, Switzerland: COLING, pp. 1284–1290. URL: https://aclanthology.org/C04-1188.

Nie, Yixin, Haonan Chen and Mohit Bansal (2018). 'Combining Fact Extraction and Verification with Neural Semantic Matching Networks'. In: *CoRR* abs/1811.07039. arXiv: 1811.07039. URL: http://arxiv.org/abs/1811.07039.

Riedel, Sebastian et al. (June 2013). 'Relation Extraction with Matrix Factorization and Universal Schemas'. In: *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Atlanta, Georgia: Association for Computational Linguistics, pp. 74–84. URL: https://aclanthology.org/N13-1008.

Santos, C´ıcero Nogueira dos (July 2015). 'Classifying Relations by Ranking with Convolutional Neural Networks'. In: *Association for Computational Linguistics*. URL: https://aclanthology.org/P15-1061.pdf.

Sawant, Uma and Soumen Chakrabarti (May 2013). 'Learning joint query interpretation and response ranking'. In: pp. 1099–1110. DOI: 10.1145/2488388.2488484.

Surdeanu, Mihai et al. (July 2012). 'Multi-instance Multi-label Learning for Relation Extraction'. In: *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. Jeju Island, Korea: As-

sociation for Computational Linguistics, pp. 455–465. URL: https://aclanthology.org/D12-1042.

Thorne, James et al. (June 2018). 'FEVER: a Large-scale Dataset for Fact Extraction and VERification'. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 809–819. URL: https://aclanthology.org/N18-1074.

Towards Data Science, Shivanee Jaiswal (2022). *Simple dependency relation between two words*. [Online; accessed March, 12, 2022]. URL: https://miro.medium.com/max/656/1*NrOGJABfU_dp2qOasutr_Q.png.

Xu, Kun et al. (Sept. 2015). 'Semantic Relation Classification via Convolutional Neural Networks with Simple Negative Sampling'. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics, pp. 536–540. DOI: 10.18653/v1/D15-1062. URL: https://aclanthology.org/D15-1062.

Xu, Yan et al. (Sept. 2015). 'Classifying Relations via Long Short Term Memory Networks along Shortest Dependency Paths'. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics, pp. 1785–1794. DOI: 10.18653/v1/D15-1206. URL: https://aclanthology.org/D15-1206.

Yahya, Mohamed et al. (Oct. 2014). 'ReNoun: Fact Extraction for Nominal Attributes'. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, pp. 325–335. DOI: 10.3115/v1/D14-1038. URL: https://aclanthology.org/D14-1038.