Open Information Extraction for Knowledge Representation

Triple Extraction and Information Retrieval from Unstructured Text

Injy Sarhan

Open Information Extraction for Knowledge Representation

Injy Sarhan

SIKS Dissertation Series No. 2023-13 The research reported in this thesis has been carried out under auspices of SIKS, the Dutch Research School for Information and Knowledge Systems.

This work was partially funded by the European Union's Horizon 2020 research and innovation program under grant agreement no. 883588 (GEIGER).

Open Information Extraction for Knowledge Representation
Cover design: Chad `https://www.fiverr.com/cebooker`
ISBN: 978-94-6469-298-3
DOI: 10.33540/1739

Open Information Extraction for Knowledge Representation

Triple Extraction and Information Retrieval From Unstructured Text

Open Informatie Extractie voor Kennisrepresentatie

Triplet-extractie uit ongestructureerde tekst
(met een samenvatting in het Nederlands)

Proefschrift

ter verkrijging van de graad van doctor aan de
Universiteit Utrecht
op gezag van de
rector magnificus, prof.dr. H.R.B.M. Kummeling,
ingevolge het besluit van het college voor promoties
in het openbaar te verdedigen op

maandag 5 juni 2023 des middags te 2.15 uur

door

Injy Abdelsalam Sarhan

geboren op 24 februari 1992
te Alexandria, Egypte

Promotoren:

Prof. dr. S. Brinkkemper
Prof. dr. M.R. Spruit

Beoordelingscommissie:

Prof. dr. A.P.J. van den Bosch
Prof. dr. C.J. van Deemter
Prof. dr. Y. El-Sonbaty
Prof. dr. W. Kraaij
Dr. S. Verberne

Acknowledgments

Reflecting on my journey at Utrecht University, I am filled with gratitude for the unforgettable and invaluable experience it has been. In my early months, I felt overwhelmed and daunted by the challenge of fitting everything into my brain, with coffee cups piling up around me. Balancing a part-time Ph.D. while working on the other side of the world and later relocating to the Netherlands amidst a pandemic was chaotic and challenging, to say the least. At times, the journey felt endless and the finalisation process seemed to stretch on indefinitely. Yet, through the stress, frustration, and occasional moments of joy, I have emerged on the other side of this transformative journey. As I take a moment to look back on the path I have walked, I feel a sense of accomplishment and pride in what I have achieved.

First and foremost, I would like to express my deepest appreciation to my supervisors, Prof. Sjaak Brinkkemper and Prof. Marco Spruit. Sjaak, I am grateful for your unwavering support and guidance throughout my Ph.D. journey. Your availability and willingness to make time for me when needed have been invaluable in keeping me on track. Thank you for being a sounding board for my ideas and for contributing to my academic development.

Indeed, the journey toward a Ph.D. can be filled with unexpected obstacles and twists, but with Marco's continuous support, I was able to overcome them and reach where I am today. I am forever grateful for the friendly working environment you have provided, which has contributed significantly to my successful completion of this work. Marco, your countless valuable academic lessons have been instrumental in shaping my research and your tremendous freedom in allowing me to choose my research directions has been empowering. Thank you for your guidance and encouragement, which have been critical in making this journey a fruitful one.

I want to express my deep appreciation to the members of the assessment committee, which comprised of Prof. dr. A.P.J. van den Bosch, Prof. dr. C.J. van Deemter, Prof. dr. Y. El-Sonbaty, Prof. dr. W. Kraaij, and Dr. S. Verberne, for their thorough review of my thesis and invaluable feedback. Their diverse expertise and perspectives have been instrumental in improving the quality of my thesis, and I am thankful for the time and effort they invested in their careful assessments. Their constructive comments and insightful sugges-

tions have helped me to refine my work and ensured its academic thoroughness.

I am deeply grateful to the members of the TDS lab for their invaluable contributions to my Ph.D. journey. Alireza, Armel, Bram, Chaim, Emil, Friso, Ian, Samar and Vincent, thank you for your willingness to engage in numerous useful discussions and for making my short time at Utrecht University enjoyable.

To Bilge, Max, Noha, and Pablo, I feel fortunate to have had the opportunity to work alongside you all. Thank you for being there during my Ph.D. journey and for being more than just colleagues. Your presence made the process less lonely, and I am grateful for the countless times you were there to lend a listening ear, offer advice, or simply share a laugh. Having you all made the experience more manageable. Thank you for being an essential part of my time at Utrecht University and beyond.

I would like to express my profound gratitude to Kees van Deemter and the members of the NLP group for the invaluable discussions and insights they shared with me. I feel incredibly fortunate to have been a part of such a stimulating and engaging academic community. Anna, Duygu, Eduardo, Juliette, Michele, and Yupie thanks for the wonderful moments we shared during our brief time together at the office. Although it was a short period, it was undoubtedly the most enjoyable and memorable time I had while you were around at Utrecht University.

Beyond the NLP group, I feel incredibly fortunate to have been surrounded by a community of amazing friends who always reminded me that there is life outside of academia, in The Netherlands and Egypt. I am grateful to have found friends who encouraged me to take a break from my Ph.D. journey and enjoy life. Though there are too many of you to name individually, I appreciate all of you who shared in delicious dinners, endless card games, and offered a listening ear when I needed to vent. Simply spending time together over a good cup of coffee was often enough to lift my spirits and rejuvenate my mind. I am truly thankful for each and every one of those moments and for the friendships that have grown from them.

Last but certainly not least, I would like to express my heartfelt gratitude to my parents and sister for their hands-off yet unwavering support. They have made tremendous sacrifices for me, and I am forever grateful for their continuous encouragement and belief in me. I deeply admire that they did not shield me from life's challenges, but rather equipped me with the resilience to face them head-on, knowing that nothing worth achieving comes easy. Thank you for everything you have taught me and for supporting every step I have taken on my Ph.D. journey. I owe my success to your unwavering love and support, and I will always be grateful for it.

I am grateful for the opportunity to have pursued my academic aspirations, and I hope to use my knowledge to make a positive impact in the world. الحمد لله، يارب زدني علماً.

Contents

1 Introduction

This dissertation introduces 'Open Information Extraction for Knowledge Representation', a systematic methodology that explores various Machine Learning (ML) and Natural Language Processing (NLP) algorithms to extract vital information from unstructured textual data to construct an effective representation of the mined information. As a part of this investigation, we focus on designing and evaluating different neural network architectures to build an effective information extraction system. All chapters together assemble the pipeline to achieve a robust knowledge representation that can benefit downstream tasks.

The field of knowledge representation deals with the question of how such knowledge bases can be constructed to ensure that the representations are highly expressive to support logical reasoning, guaranteeing that the knowledge is made available as needed (Van Harmelen et al., 2008). Our research in information extraction and retrieval follows a solution-driven methodology that builds on and contributes to existing information extraction technologies, emphasizing their application in knowledge representation.

To benefit the understanding of this dissertation, we will first introduce the technology employed in this thesis (§ 1.1), where we will scope our research to the field of NLP (§ 1.1). We will then elaborate on various Information Extraction (IE) approaches, to which methods and techniques are used and contributed to, with the aim of extracting specific information from textual sources (§ 1.2). This is followed by an overview of knowledge representation, including a discussion of the ongoing challenges in this field (§ 1.3). Then we introduce the main research question and related research questions, the main research methods that were used to investigate them, and outline the remaining chapters of this dissertation (§ 1.4 and § 1.5). Finally, we list the outcomes of this dissertation (§ 1.6).

1.1 Research Context

We live and work in an era dominated by technology, and it dominates our daily conduct of human life. Be it through intuitive search results, personalized recommenders, or voice assistants implemented in cognitive computing

systems, the common key in these applications is knowledge representation and reasoning. Knowledge representation is used to portray information from the real world, fundamentally, it is the study of how intelligent agents' beliefs, intentions, and automated thought judgments can be adequately represented to utilize this knowledge to solve complex real-life problems (Davis, 2015).

As tons of data from different sources are generated daily, research in knowledge representation aims at enhancing ways to present this information so that it could be of the highest benefit for the abovementioned applications. In this regard, researchers design, implement and evaluate various information extraction tools to extract vital data and aggregate the collected data from a specific domain into a single data structure that serves various scientific fields such as question answering systems and recommender systems. These methods and solutions are employed to add value to the enclosed data and make it reusable in meaningful and intellectual ways.

The use of ML and NLP techniques to automatically construct different architectures of knowledge representation has attained success in the last decade. Nonetheless, despite the progress made, further improvement is imperative (Davis, 2015). There still exists substantial gaps in how knowledge is represented, namely in selecting a suitable data structure to effectively express the desired information for a specific domain.

From an entirely computational perspective, knowledge representation design focuses on achieving specific aims to ensure efficient reasoning support, this includes accuracy, robustness, breadth of scope, and articulacy (Davis, 2015). Conceptually, a proper knowledge representation should cover all relevant aspects of a particular domain. The represented knowledge must be able to express the forms of partial and incomplete knowledge that occur naturally in the intended application. Further, the enclosed information should be unambiguous and well-defined as far as the topic allows. Because studies often aim to ensure that the knowledge covered is irrefutable, Pearl (2014) and Halpern (2017) argue that for a knowledge base to be beneficial for reasoning modules, it should not be restricted to information with a high level of certainty, it should instead support relative degrees of certainty. Such objectives need to be addressed while constructing the relevant knowledge representation architecture. The individual chapters of this dissertation all intend to contribute to the goal of achieving an efficient knowledge representation.

Machine Learning in Brief

A core discipline of artificial intelligence (AI) is ML, a branch of computational science whose aim is to build methods by leveraging data to interpret

and analyze patterns with minimum human intervention (Mitchell, 1997). ML makes heavy use of statistics, pattern recognition, knowledge discovery, and data mining to create algorithms that are utilized in a variety of fields such as medicine and computer vision, where it is difficult to address using traditional algorithms (Hu et al., 2020). The concept of ML came into the picture back in 1950 when Alan Turing, one of the most influential computer scientists of the 20th century, published an article to address the question *'can machines think ?'*, where he proposed a hypothesis stating the machines that succeed in convincing humans that it is not indeed a machine, then it would have achieved AI, this was known as the Turing Test (Turing, 1950). Later in 1959, ML researcher Arthur Samuel defined ML as *'a field of study that gives computers the ability to learn without being explicitly programmed'* (Wiederhold and McCarthy, 1992). Despite being highly publicized nowadays, neural networks date back to the 1950s when Frank Rosenblatt — inspired by the human thought process — designed the first neural network for computers, commonly called the perceptron model (Tappert, 2019).

In the 1990s, the concept of deep learning methods was coined, as the large data availability contributed to the shift of ML from a knowledge-driven approach to a more data-driven approach, in which a large amount of data is analyzed, and machines learn and draw conclusions from the results. Since then, as large datasets were made available several variants of deep learning as deep neural networks and deep reinforcement learning were able to further enhance state-of-the-art results at different tasks (LeCun et al., 2015). Recent breakthroughs in this field include IBM's Watson, which relied on ML along with information retrieval techniques to defeat two human champions in a game of Jeopardy in 2012 (Ferrucci, 2012).

ML has scaled exponentially in the past decade as the quantities of the data produced continued to grow, and so did the computer's ability to process and analyze data. Emerging computing technologies have turned AI and ML potential game changers into proper drivers of innovation. ML and business intelligence are empowered by data as enterprises progressively depend on unstructured data to effectively monitor their businesses for analytical and decision-making purposes (Delen, 2014).

The primary complication is that the majority of this data is in an unstructured format, thus limiting the analysis and insights that can be drawn from such a format. The vast amount of unstructured text motivated computer scientists and artificial intelligence practitioners to develop NLP tools to analyze the data effectively and efficiently. In this work, we apply traditional supervised learning in Chapter 5, in addition to employing an unsupervised clustering algorithm in Chapter 6. We explore deep learning methods in Chap-

ters 3, 4, 5 and 6.

Setting the Scene in NLP

Based on constructing computational techniques, NLP permits automatic analysis and interpretation of textual information. Due to the exponential growth of unstructured texts, the emergence of NLP-based techniques has empowered us to easily analyze relevant information using various systems, such as instant translation, smart assistats, and search engines. This has not been trivial to achieve. For many years, long-established research has been directed to bridge the gap between ML and NLP.

NLP is thriving across many domains, all of which are highly adaptable in the industry verticals such as healthcare, news, and financial domains. The fact that retrieval and thus the representation of data can be enhanced conveys that advancement in NLP permits the users to be better understood across various areas. Whether phonology, pragmatics, morphology, syntax, or semantics, NLP is fast-paced in all areas to keep up with the requirements of today and the future.

Several tasks in NLP, such as Information Retrieval (IR), Named Entity Recognition (NER), Relation Extraction (RE), and Open Information Extraction (OIE), all fall under the umbrella of Information Extraction (IE), in which the goal is to extract explicit properties from textual data along with the relationship that is enclosed between tokens. This dissertation aims to design IE systems for harvesting valuable information from unstructured textual data, emphasizing the need for a robust knowledge representation for efficient and effective retrieval and understanding of data.

NLP Through Time

The traditional NLP methods for several tasks involve training a standard classification algorithm on a rich set of manually-crafted features that are extracted from the input sentences. The components that make up the features start as an empirical process. For instance basic text processing features include; tokenization, stemming and lemmatization to train semi-supervised pattern-based models. By primarily relying on linguistic intuition, observations drawn from initial experiments evolve the processes to produce a feature selection that is task-dependent, paving the way for additional research for each new NLP task. Figure 1.1 depicts a timeline of the development of the most commonly used NLP methods that are employed in this thesis. We briefly discuss each breakthrough in the upcoming paragraphs.

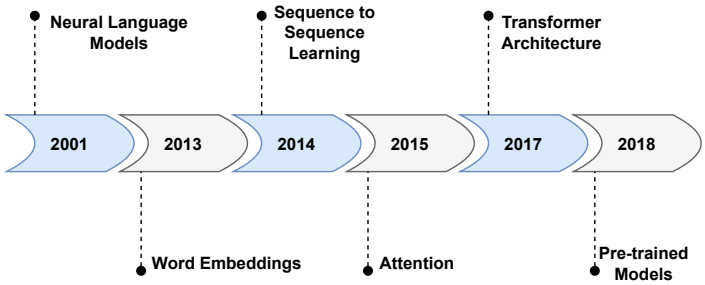

Figure 1.1: NLP development timeline.

Language modeling is the task of predicting the next word in a given sentence using the previous words. This is perhaps the most straightforward language processing task with concrete functional applications such as predictive keyboard and spelling auto-corrections (Kannan et al., 2016). Traditional language model approaches are based on n-grams and utilize smoothing to handle with unseen n-grams (Kneser and Ney, 1995). Neural networks have been used for language modeling since the 2000s. In 2001, (Bengio et al., 2001) proposed the first neural language model consisting of a feed-forward neural network with hidden layers, the authors point out the curse of dimensionality caused by the large vocabulary size of natural languages that results is complex computations. They design a feed-forward neural network that jointly learns the language model and vector representations of words.

In the 2010s, deep neural networks paved their way to the NLP territory, starting with a simple feed-forward artificial neural network (Lavine and Blank, 2009), that are utilized to learn the relationship between independent variables, which is later fed as input to the network. A fully connected feed-forward network forms a Multi-Layer Perceptron (MLP), which is the start of Deep Neural Networks (DNNs). The real power of deep learning comes from the ability to learn features from the data, instead of relying on hand-built human features for classification, representations like word embeddings are utilized to train the networks. This later progressed to more evolved variations, Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) (LeCun et al., 2015). CNNs can comprehend spatial relations, in which they have the capability to develop an internal representation of a two-dimensional image, which makes it a perfect fit for classification tasks on images or videos. Given their supremacy in the computer vision domain, CNNs are also employed in the field of NLP, however, their limitation in processing sequential data makes

them non-ideal for textual tasks.

RNNs made their impact in the NLP world due to their 'recurrence' feature that makes them suitable for sequential and temporal data (Goodman et al., 2014). That being said, RNNs suffer from unstable behavior due to vanishing and exploding gradients (Pascanu et al., 2013). As the back-propagation advances backward from the output layer towards the input layer, the gradient diminishes drastically till it approaches zero, this hinders the process of knowing the directions in which parameters should move to enhance the cost function, as the weights of the lower layers remain almost unaffected resulting in the gradient descent never converging to optimum. Furthermore, large error gradients may accumulate resulting in large updates to the weights during training, the learning process is compromised causing the networks to be unstable leading to poor prediction models, this is known as exploding gradient. Several techniques were used to mitigate vanishing and exploding gradients, such as gradient clipping (Pascanu et al., 2013), where the gradient is clipped during back-propagation to a certain threshold so that it never exceeds the specified value, guaranteeing that the gradient will not explode.

Employing Long Short-Term Memory (LSTM) networks (Hochreiter and Schmidhuber, 1997) alleviates the exploding and vanishing gradient problems despite the depth of the network, as the neural network is able to remember the gradient values. The forget gate in the LSTM network controls which information to forget and which to retain, making LSTM an appropriate choice for learning long-range dependencies.

A variant of LSTM is Gated Recurrent Units (GRUs) (Chung et al., 2014), which is deemed as a less-complex and faster version of LSTM as it merges the gating functions of the input and forgot gates into a simple update gate. In addition, it combines the cell state and the hidden output into a single hidden state layer making the training process relatively fast. Since the associated gates are able to specify which values to update at each time-step, therefore not all weights get affected at the same time for each step in back-propagation, this is how GRUs are able to eliminate the problem of vanishing gradients.

Sequence-to-sequence (seq2seq) model is a special class of RNN architecture that was introduced in 2014 by Google (Sutskever et al., 2014). Seq2seq learning aims to map a fixed length input with a fixed length output, however, the length of the input and output may differ, and thus it is commonly used in machine translation. The encoder-decoder is the most common architecture in seq2seq models, where the encoder reads the input sequence and utilizes the hidden state to summarize and encapsulate the information, and outputs a single vector that is sent to the decoder to produce an output sequence (Cho et al., 2014; Sutskever et al., 2014).

Word2vec (Mikolov et al., 2013a) revolutionized NLP approaches when it was introduced in 2013. As the name applies, Word2vec presented a way to denote similarity and relationships between words through the use of word vectors. Thus, enabling the extraction of insightful information. This inspired researchers to dive deeper into vector representations to bring out the semantic similarity of words. Several derivatives of Word2vec were later introduced such as Global Vector word representations (GloVe) (Pennington et al., 2014), the main difference between the two models is the training method. Word2vec is based on training a shallow feed-forward neural network using either the skip-gram or Continuous Bag of Words (CBOW) model. The former predicts the context words via a hidden layer that takes each word in the corpus as an input, while the latter obtains the embedding of a specific word from the hidden layer by taking the context of the word as an input to the model. On the other hand, GloVe is based on matrix factorization techniques. In both models, cosine distance is deployed to represent the relationship between words, however, word representations obtained from both approaches do not take context into consideration.

Distributional word representation later evolved beyond static vector representation to dynamic contextual embeddings that resulted in ground-breaking performance across several NLP tasks. The word features obtained from such models enable each obtained word vector to be associated with a representation that is a function of the entire input sequence, hereby acquiring semantic and syntactic features of words under diverse linguistic contexts (Liu et al., 2020a). Earlier versions of contextual embeddings relied on LSTM networks such as Embeddings from Language Model (ELMo) (Peters et al., 2018) and Universal Language Model Fine-Tuning (ULMFiT) (Howard and Ruder, 2018). Subsequent language models improve parallelization by generating embeddings from a so-called transformers architecture, where the vectors are generated by passing the entire sequence to a pre-trained model. The transformer-based models rely on the attention-mechanism, in which a word is related to its neighbors in a specific way.

Where is NLP currently ?

"*Every once in a while, a revolutionary product comes along that changes everything.*" This quote by Steve Jobs (2007) perfectly describes the effect that the attention mechanism had in the field of NLP. The core concept behind attention is that the output of the model is predicted based on the most relevant parts of the input sentence. This is achieved by developing an interface that connects the encoder and decoder components, in a way that enables the

decoder to obtain information from every encoder's hidden state.

In this day and age, we are experiencing a quantum leap in the way we develop deep learning models owing to the attention mechanism, it has procreated the rise of transformer architectures in NLP (Vaswani et al., 2017). Self-attention is the new cutting edge in transformer-based models. Google's Bidirectional Encoder Representations from Transformers (BERT) achieved state-of-the-art results across various NLP tasks by applying bidirectional training of the transformers to language modeling (Devlin et al., 2019a). The bidirectional capability enables BERT to process text from both directions at the same time. Several adaptations later followed such as XLNet (Yang et al., 2019) and XLM-RoBERTa (Conneau et al., 2020a). BERT is trained on two different, yet related NLP tasks: Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). In the former task, the model predicts the masked word, while in the latter the model decides whether a relationship exists between a pair of sentences or not. Later models, such as XLM-RoBERTa and Efficiently Learning an Encoder that Classifies Token Replacements Accurately (ELECTRA) (Clark et al., 2020), were only trained on MLM as training on NSP was deemed unnecessary.

In this thesis, we explore several neural approaches to design and demonstrate an information extraction system and ensure that our model is reliable and robust. To build models that go to production, we must take into consideration the computation power needed along with the performance, as neural models can impact the computational cost which is considered significant for large-scale applications or applications requiring real-time response.

1.2 Information Extraction

Day after day, a vast amount of textual data is produced and published. Indeed, it is a highly complex task to extract valuable information from unstructured data manually. Information Extraction (IE) techniques are developed to decipher natural languages to extract valuable data from unstructured text. Consequently, it is vital to develop automatic IE systems; otherwise, a considerable quantity of valuable information might remain obscure if we are incapable of retrieving this data. In an effort to benefit from the increasing amount of unstructured text and to simultaneously overcome the challenges of presenting the extracted information in a beneficial way to bridge the gap between IE and downstream applications, the primary focus of this dissertation is to design and evaluate different IE systems.

Being the process of extracting information from unstructured textual sources, IE is deployed to find and extract entities, in addition to classifying and storing them in an accessible knowledge base. Semantically enhanced IE couples the extracted entities with their semantic descriptions, such as metadata about concepts connected to an existing ontology. By appending metadata to the mined entities, IE is further enriched and can overcome several challenges in knowledge discovery. Semantically tagged documents are not only easier to interpret but also become more sustainable and facilitate searching as it defines the textual data via linking the references to concepts and entities stated in the corpus. This section briefly discusses the primary IE techniques exploited in this dissertation.

Information Retrieval

Regardless of the data type, whether it is text, images, or videos, Information Retrieval (IR) is a crucial part of the data analysis procedure. Being the foundation of search engines, IR is the main component that sustains the extraction of information from textual data. There are many strategies to digitally pre-process documents to accelerate the retrieval procedure. These strategies are categorized under the general concept of indexing. Lexicon-focused word indexing, in which terms in the index are mapped to individual words in a document, results in inaccurate and incomplete retrieval, thus a more reliable indexing semantics-based scheme that relies on concepts instead of keywords is utilized in more current IR systems where the phrases are matched to an existing thesaurus, thus alleviating the synonymy challenge and improving the recall. As retrieval is performed in the conceptual space and accurate results are retrieved even in the case of ambiguous terms, it also decreases the extraction of false positives, which resolves polysemy problems and enhances the precision (Egozi et al., 2011). Since documents and queries are embodied as a set of lexical entities, conventional IR models are grounded on CBOW representation. Additionally, Term-Frequency Inverse Document Frequency (TF-IDF) — a numerical statistic technique that is used to reflect the relative importance of a term in a document — is often deployed as a weighing methodology for the extracted documents (Boubekeur and Azzoug, 2013).

Named Entity Recognition

Named Entity Recognition (NER) classifies standard concepts, whether general concepts such as people, organizations, numbers, or domain-specific concepts in fields like cybersecurity and bio-sciences (Tjong Kim Sang and De Meul-

der, 2003). Semantic annotation is later deployed to disambiguate the identified entities such that they are unambiguously identified according to a field-specific knowledge base. The labeled terms are linked to the broader context of already existing data by identifying and converting the annotated text chunks into machine-comprehensible data pieces. Conventional NER systems can be categorized into three primary streams: (1) Rule-based systems that don't necessitate labeled data as they depend on hand-crafted patterns; (2) Unsupervised learning systems; (3) Feature-based supervised learning systems.

The first category of NER systems uses hand-crafted rules to automatically produce extraction patterns for entity identification. For instance, the use of grammatical properties such as Part-of-Speech (POS) and syntactic features to structure patterns and locate entity boundaries (Sari et al., 2010). An example of a pattern mining technique, mXS, was introduced in (Nouvel et al., 2011). mXS performs NER by comprehensively searching for hierarchical sequential patterns through identifying annotation rules in a training corpora while continuously evaluating the quality of the patterns.

Unsupervised learning systems depend on clustering without the need of having annotated data. Several approaches were proposed ranging from a stepwise methodology to classify entities in and detect entity boundaries using a noun phrase chunker, an inverse document frequency filter, and a classifier dependent on distribution semantics (Zhang and Elhadad, 2013), to merge disambiguation with named entity extraction, which has a core advantage that it can adapt to any named entity type.

The critical concept in supervised NER systems is to learn features of both positive and negative named entities from a labeled corpus by designing patterns and rules that capture occurrences of the predefined entities (Nadeau and Sekine, 2007). A wide range of ML techniques are employed in a supervised environment, such as Hidden Markov Models (HMMs) algorithms to classify named entities. Such models often deploy chunk taggers to construct a NER system that is built on the mutual information independence assumption as an alternative to conditional probability (Zhou and Su, 2002) or by a model allocating the joint probability to label sequences and then rely on learning by example methodology (Morwal et al., 2012).

The Conditional Random Field (CRF) is a statistical model that is particularly effective in addressing NER problems as it takes context into account. In other words, when making predictions, the CRF model incorporates the influence of adjacent data points by representing the prediction as a graphical model (Sutton and McCallum, 2012). The linear chain CRF, a common type of CRF model, assumes that the tag for the current word is determined solely by the tag of the preceding word. This is somewhat similar to HMMs, al-

though the CRF's topology is an undirected graph. A limitation of the linear chain CRFs is that they are only able to account for label dependencies in a forward direction (Yu et al., 2010). For instance, if the model comes across an entity like "New York University", it may identify the token "York" as a name because it cannot consider the downstream occurrence of the "University" token. One solution to this issue, as elaborated in Chapter 6, is to incorporate a bidirectional RNN between the input words and the CRF layer. For more details on recent state-of-the-art NER-CRF approaches, please refer to Section 6.2.

Recent advances in DNNs have enhanced the performance of NER systems, and broad-spectrum research to explore several different network architectures was carried out. Such techniques include designing a CNN for extracting character-level representations in which the output is then fed to an RNN context encoder (Ma and Hovy, 2016) or a bidirectional recursive network (Li et al., 2017b). In addition to other RNN-based models, such as employing a stack residual LSTM and trainable bias decoding, where word features are extracted from word embeddings (Tran et al., 2017). NER systems later evolved to deploying attention-based mechanisms and pre-trained language models (Naseem et al., 2021; Luo et al., 2018). In this dissertation, we will elaborate on how NER can complete the puzzle of knowledge representation.

Relation Extraction

Relational Extraction (RE) builds on top of NER and plays a significant role in the field of Natural Language Understanding (NLU) to automatically predict and extract semantic pre-defined relationships that associate entities of a certain type such as Person, Organization, or Location in a given sentence. The relationship can fall into a number of semantic categories, for instance, Born-In, Lives-In, Employed-By. An example is the extraction of an 'Is-In' relationship from the sentence "London is in England" states the presence of a relation between the two entities London and England. This can be denoted using triples, $\ll$ *London, is in, England* $\gg$.

One might need to extract interactions between genes and diseases or drugs and diseases to construct a clinical ontology or to simply find relationships among people to develop an easily browsable database. The abundance and heterogeneity of unstructured text in any field are deemed a challenging process to be processed manually or even in human-in-the-loop models. Thus, the extraction process is fully automated.

Conventional non-deep learning RE methods commonly operate in the su-

pervised paradigm. Early approaches can be sub-categorized into two methods which are feature-based approaches and kernel-based approaches. The kernel method uses a linear classifier to address a non-linear problem. This is carried out by transforming the data to a linearly separable one. RE is usually tackled in a classification manner using various features as elaborated in the research carried out by Hendrickx et al. (2009) such as POS and dependency parse features. Both approaches partially rely on auxiliary NLP systems resulting in downstream error propagation. Additionally, having a relatively large quantity of labeled textual data is indispensable in supervised approaches to accurately train an RE model, which poses restrictions due to computational resources, time, quality, and cost constraints on the labeling activity. Such traditional rule-based methodologies might not capture all the essential information to extract, resulting in a moderate recall (Wang et al., 2022). Semi-supervised or bootstrapping approaches are more lenient regarding the amount of labeled dataset needed, as extractions can be done by using the patterns generated from a small number of seed pairs (Brin, 1999). While distant supervision in RE relies upon the co-occurrence technique, it implies a relation if entities are mentioned together. That is typically carried out using distance-weighted count and frequency normalization. Hence, a major advantage of weak supervision methodologies is that it does not necessitate labeled data, it only requires access to an existing knowledge base. However, it may yield many false positives as related entities may co-appear in a sentence without theoretically expressing a relation (Junge and Jensen, 2020). To alleviate the concern mentioned above and lessen the noise, Riedel et al. (2010) moderated the distant supervision hypothesis by modeling the task as a multi-instance learning problem that assigns a label to every bag of entities, under the assumption that "if a relationship exists between an entity pair, then at least one document in the bag for the entity pair must reflect that relation".

Continuous evolution in deep learning addresses many of the prior mentioned problems. As RE requires the relation between entities to be predetermined, making the process of acquiring labeled examples harder, transfer learning is a sensible solution when it comes to dealing with the limited availability of labeled data. Deep learning approaches are either structure-oriented or semantic-oriented. The former enhances the ability of feature extraction through amendments in the architecture of the model. The latter boosts the ability of semantic representation by mining the internal association of the sentences (Wang et al., 2022). In this dissertation, we investigate how transfer learning can aid in the RE task.

Open Information Extraction

Open Information Extraction (OIE) is the task of extracting relation triples directly from unstructured text. The extracted triples enclose relevant entities that can be subjects or objects and relationships in the form of predicates.

Nevertheless, unlike RE, OIE is performed without the need for an explicit ontology, pre-defined relation, or schema, thus enabling the extraction of a broader range of valuable data that can be found in a piece of text. This is deemed crucial with the explosive growth of open-domain and diverse unstructured textual data. The extracted triples can be utilized as the primary source data for several downstream tasks, including question answering, summarization, and knowledge base population (Smith et al., 2022).

Several works have been proposed in OIE in the past, ranging from extraction using self-supervised classifiers to pattern-based and semi-supervised approaches. In recent years, there have been significant breakthroughs in using neural network methods for OIE. For instance, the performance of CNN, LSTM, and transformer-based models were able to surpass traditional approaches (Van Le et al., 2022). Naturally, as with the previously mentioned IE systems, each OIE approach is associated with quite a few shortcomings that prevent the OIE task from being solved. These include incoherent extractions that result from pattern-based models, in which the extracted triples have no meaningful interpretation. Another problem is overly-specific relations that convey too precise triples to be beneficial in further downstream semantic tasks (Niklaus et al., 2018a). An ongoing challenge of triple extraction from a heterogeneous dataset is the extraction of uninformative triples, which is inevitable and often leads to noisy labels and distorts. Different OIE approaches, along with their respective drawbacks, are discussed in detail in the following chapter. All things considered, OIE remains an active research topic, owing to its ability to mine information for domain-independent usage.

Relation Triples in Open Information Extraction

Relational triple extraction involves the automatic identification of semantic relations that exist between multiple entities in a sentence using triple structures comprising a subject, relation, and object. This step is critical in constructing knowledge graphs from unlabelled text corpora, as emphasized by Dong et al. (2014). In this section, we provide a more detailed explanation of the notion of triples in our research.

In the context of OIE, extractions are represented as a tuple in the form of $\ll$ *Entity1 - Relation - Entity 2* $\gg$. In the discussion on RE in Section 1.2,

we introduced the concept of triples and their significance in RE. However, it is important to note the differences between RE and OIE, which are further explained in Chapter 4 (refer to Table 4.1 for an example of the differences between both extraction methods).

Due to the relation-independent extraction process in OIE, it becomes challenging to utilize the complete set of features that are typically employed when performing extraction on a single relation. For example, as demonstrated in the work of Banko and Etzioni (2008), when presented with the sentence "*Microsoft is headquartered in beautiful Redmond*", we expect to extract ≪ *Microsoft - is headquartered in - Redmond* ≫. The presence of terms such as "company" and "headquarters" can aid in identifying instances of the "headquarters"(X, Y) relation, but they would not contribute to identifying relations in general. Furthermore, named entity types are often used in relation extraction to guide the process (e.g., the second argument in "headquarters" should be a "location"). However, in OIE, the relations themselves are not known beforehand, as stated by Banko and Etzioni (2008).

The unique nature of the OIE task has motivated us to further investigate and enhance this promising field. In order to provide a more detailed understanding of the extracted triples, we classify our extractions into the following categories:

- Binary triples: a normal triple extraction without any overlapping entities. Example: "*Nelson Mandela was the President of South Africa from 1994 to 1999.*" ≪ *Nelson Mandela - was the President of - South Africa* ≫.

- Individual overlapping entity triple: a type of extraction where one of the entities is identical to an entity in another triplet. For example, in the sentence: "*Barack Obama, who was born in Honolulu, the capital of Hawaii, served as the 44th President of the United States.*" The triplet ≪ *Barack Obama - born in - Honolulu* ≫ overlaps the entity "*Barack Obama*" in ≪ *Barack Obama - served as - the 44th President of the United States* ≫.

- Pair overlapping entity triple: triplets that share a pair of overlapping entities. For instance: "*John Smith, the founder and CEO of XYZ company, has played a crucial role in its success.*" The triplets ≪ *John Smith – founder - XYZ company* ≫ and ≪ *John Smith -, CEO of - XYZ company* ≫ share the entity pair "*John Smith*" and "*XYZ company*".

While our proposed OIE methodologies are capable of covering a wide

range of extractions, there are some types of extractions that they do not cover. These include:

- N-ary extraction: the extraction of relations with more than two arguments, such that relationships between entities need to be captured in a more granular and detailed manner. For example in the sentence "*The movie stars Tom Cruise and Miles Teller"* the extraction would be represented as a 3-ary relation, where there are three entities interconnected by the "stars" relationship. The extraction would be ≪ *The movie – stars – Tom Cruise - Miles Teller* ≫.

- Nested extraction: involves the extraction of relations within other relations. For example in the sentence "*The company acquired a startup that develops AI-powered chatbots.*" ≪ *the extraction would involve a nested relation where "acquired" is the main relation and "a startup that develops AI-powered chatbots" is an argument that itself contains a relation. The extraction would be represented as:* ≪ *the company - acquired -* ≪ *a startup, - develops - AI-powered chatbots* ≫ ≫. By leveraging nested extractions, we can capture more complex relationships and patterns of behavior in natural language text.

- Implicit relation: refers to sentences where the relationships between the entities are not explicitly stated but can be inferred based on the context or the knowledge of the reader. For example, in the sentence "*The CEO of Google, Sundar Pichai, announced new products at the conference*", an explicit extraction would be ≪ *Sundar Pichai – announced – new products at the conference* ≫. However, the model may miss the implicit relation≪ *Sundar Pichai – works at – Google* ≫. For further information on OIE for implicit relations, please refer to (Beckerman and Christakis, 2019).

By recognizing the limitations of our proposed OIE methodologies, we can better understand the scope of the task and identify areas for future research and development as discussed in the Future Work section of Chapter 7.

Openness in Information Extraction: Explanation and Implications

Conventional IE methods focus on addressing specific, clearly-defined inquiries over a pre-determined collection of target relationships within limited, uniform corpora. As a result, transitioning to a new domain necessitates that the user not only identify the target relations but also manually establishes new extraction guidelines or hand-annotates additional training data (Niklaus et al.,

2018a). This highlights the significance of the 'openness' principle in OIE. The openness implies that extractions are performed on open-domain sentences, encompassing any subject or domain. In contrast to RE, as elaborated in Section 1.2, OIE does not depend on domain-specific knowledge or predefined schemas for information extraction. Rather, its distinguishing feature from other IE systems lies in its capacity to handle text from any source, making it highly adaptable and versatile.

Extracting domain-agnostic information presents some difficulties, particularly with adaptability. OIE systems need to be versatile and capable of handling a wide range of domains, making the implementation of these systems more complex compared to traditional IE systems. The design choice might lead to a degradation in the quality of the extractions and lead to either uninformative triples or low information coverage. Thus, maintaining a high-quality model is a critical challenge. These issues are further examined and addressed in Chapters 4 and 6 of this dissertation.

This research focuses on the deployment of automatic learning techniques to overcome the core weaknesses of the application of IE and its inherent reliance on a domain by lowering the demand for supervision. Explicitly, throughout this dissertation, several IE models are developed using various neural architectures to build robust domain-agnostic methods for the automatic extraction of information. Further, we experiment with the transferability of these models across domains and semantically-related IE tasks.

1.3 Knowledge Representation

"*We are drowning in information and starving for knowledge*"– (Jain et al., 1985).

Extracting information from data does not entail obtaining knowledge. Knowledge representation and reasoning are vital areas in AI, as they are directly associated with AI agents' intelligent behavior, such as chatbots and expert systems. Knowledge is rather seen as a dynamic component, and it has to constantly change and improve as it is accountable for representing information about the real world. The challenge is to convert the extracted information from text into knowledge that can match the user's standards and enrich the reader with the understanding they desire. That is why knowledge representation goes hand in hand with automated reasoning, as the primary aim in portraying knowledge is the interpretation ability associated with that knowledge.

Being a core component in both expert and recommender systems, a knowl-

edge base encloses all relevant information to represent facts and rules that a non-expert user might need in a specific domain. A key component in building such systems is knowledge acquisition and learning modules. This component's role is to permit the expert and recommender systems to attain more information from several sources and feed it into the knowledge base. Thus, making the trade-off between expressivity and practicality one of the most important design aspects to consider when it comes to knowledge representations.

Classifying Semantic Networks

As semantics and linked data become ever more mainstream, it is essential to comprehend the difference between different types of semantic networks to make the correct decisions regarding metadata management, as it directly influences how one deals with this data and its deployment in downstream tasks. In the upcoming paragraphs, we briefly discuss distinct types of data representations and how they relate to the work carried out in this dissertation.

A taxonomy's definition is intuitive; it is a formal structure of classes or types of objects within a domain that are used to provide machines ordered representations. Taxonomies are valuable tools for content organization and retrieval as they yield helpful input for semantically intensive tasks. Generally, a hierarchical relationship can be portrayed as any asymmetrical relation that defines subordination between two nominal terms (Bordea et al., 2016). Taxonomy learning from unstructured text is a challenging task that can be categorized into numerous subtasks, such as term extraction, taxonomy induction, and classification. Prior to constructing a taxonomy, it is crucial to verify the lexical relationships between words, and this phase is often overlooked and a relatively less well-studied topic. Due to the acknowledged importance of this phase, in Chapter 5, we focus on taxonomy classification and regression of relationships that hold between nominal arguments in a sentence.

In the early 2000s, ontologies conquered the AI field as they became a fundamental building block in search engines. Although the two words 'taxonomy' and 'ontology' have very similar meanings, the latter is more theoretical and requires more advanced elements such as relationships between concepts to permit formal reasoning about the meanings of sentences and hereby overcoming the limitations of taxonomies. To elaborate, ontologies are a knowledge representation formalism in which critical terms in a particular field, along with their properties, relations, and restrictions, are formally established (Riaño et al., 2019). Ontologies stand firm when it comes to their compatibility with inference reasoning tools that can be utilized to deduce new knowledge based on a set of principles. On the other hand, they do not capture the data re-

lated to the set of axioms, and this is where ontologies end and the Knowledge Graphs (KGs) begin.

A KG stores facts or extracted information triples that are constituted of ≪ *subject, predicate, object* ≫, where the predicate represents the relation between the extracted entities. Essentially, KGs are designed with a rigid ontology to support their semantics and emphasize the essence of knowledge representation (Nguyen et al., 2020). Rather than being integrated into a document, semantic metadata might be saved in a KG. The annotations can be stored as distinct objects that refer to the document, which is also a node in the graph that allows for a wide range of analytics. Thus, permitting the annotation to be indexed and queried alongside other types of data.

When creating a KG, the breadth of information sources used might lead to redundancy, semantic diversity, and non-uniform quality that results from the heterogeneous information from multiple sources. That is why it is essential to carry out operations like conflict detection, entity disambiguation, and entity fusion to effectively fuse knowledge to produce a large-scale and high-quality KG. Resolving this issue is a non-trivial task for several reasons. To begin with, capturing the contextually rich dynamic semantic correlations between features has its difficulties, as the process of integrating semantic correlations between terms to generate complete, concise, and interpretable semantic explanations is far from being resolved. Additionally, noisy and uninformative data are easily extracted from the unstructured text as the search space exponentially grows and thus becomes more complicated (Ayranci et al., 2022). To combat these challenges, we aim to design an efficient knowledge representation capable of adequately capturing semantic correlations.

Addressing the issues mentioned above requires fusion and canonicalization techniques. These techniques deal with the consequences that arrive from the high volume and heterogeneity of the data that is used to populate the KG to achieve a cohesive perspective. Entity linking is one of the traditional tactics to canonicalize noun phrases by mapping them to an existing knowledge base, as Wikipedia uses the help of a ranking system (Lin et al., 2012). However, a drawback associated with these approaches is that noun phrases can refer to non-existing entities in the knowledge base. Thus research on how to deal with the challenge of clustering such entities is actively ongoing. An example of one of the proposed solutions by (Yates and Etzioni, 2009) is relying on string similarity-based features to cluster the entities, but it is still unsuccessful when handling synonymous phrases. While Vashishth et al. (2018a) canonicalize KGs using *side information* for both, the noun and relation phrases, to avoid storing redundant and ambiguous facts in the KG. Such side information includes using Wordnet (Miller, 1995) for word-sense disambiguation,

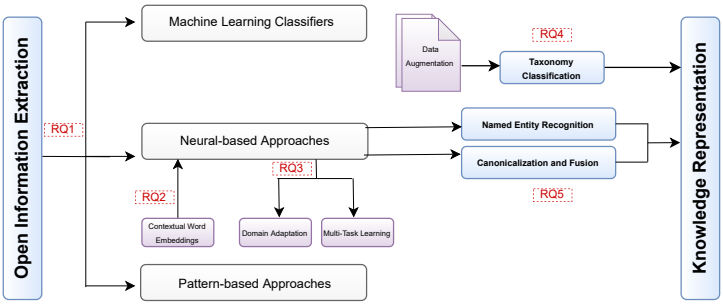

Figure 1.2: The conceptual model of this dissertation, along with a mapping of the five Research Questions onto these phases. Abbreviation: RQ = Research Question.

IDF for document overlap, and morphological normalization operations such as capitalization and pluralization to search for equivalent noun phrases. All these, together, contribute to a canonicalized KG.

Conventional IE methodologies to construct KGs incline towards using a pre-defined ontology or schema that encloses a selected collection of well-known concepts in a specific domain. Nonetheless, it is problematic in some domains when a dedicated schema is unavailable. As a result, in Chapter 6 of this dissertation, we dive deep into knowledge representation by experimenting with deploying OIE methods for building a KG that does not necessitate a pre-defined set of relations. Since we are relying on OIE and openly extracting relations, it might lead to an explosion of redundant information, which is undesirable, as it will not only take up more memory but also certainly produce sub-optimal outcomes. Consequently, we apply fusion and canonicalization techniques to ensure an efficient querying experience for the users.

1.4 Research Questions

Although humans can easily learn a language, the ambiguity and inaccurate characteristics of natural language make it problematic to implement NLP on a machine, making the process of extracting information and knowledge representation relatively complex. Extensive research is still needed in this field to achieve interpretability in open KGs.

The conceptual model in Figure 1.2 visualizes the five stages of the OIE system proposed in this dissertation, each corresponding to a Research Ques-

tion (RQ). The first is domain understanding different OIE methodologies and identifying the problem being solved (RQ1). After RQ1 shows which machine learning OIE models are the most promising, we follow by prototyping a neural OIE model in the second phase to validate the requirements and confirm the system's behavior by investigating the performance of different word embeddings (RQ2). In the third phase, we highlight the effectiveness of transfer learning of OIE by extending the work of our proposed framework in the second phase to a different domain and a related NLP task (RQ3). While in the fourth phase, we start exploring the knowledge representation part by classifying the taxonomic relationship that binds nouns in a sentence; this is mainly done using data augmentation techniques (RQ4). Conclusively, in the fifth and final phase, we assemble the pieces and employ OIE, NER, and fusion techniques to build a KG (RQ5). Throughout the dissertation, we investigate ways to improve the performance of OIE systems for the sake of populating the KG from unstructured text with high-quality triples. This constitutes exploring advanced techniques to perform canonicalization over the extracted OIE triples. Therefore, we pose the following main research question:

MRQ — To what extent can we enhance open information extraction methods to efficiently and effectively represent unstructured textual data?

To address this question, we design an Open Information Extraction (OIE) framework that can be used as the primary building block to populate KGs, which results in an efficient data representation and hereby facilitates querying by users. The aim of this research is threefold. First, our main aim is to investigate different OIE approaches and improve their performance. Secondly, we aim to use the extracted information from the OIE model to build a KG without the need for an existing knowledge base or an ontology. Finally, as a result of building an open KG, we apply effective fusion and canonicalization techniques to alleviate the effects of redundancy and conflict detection. This eventually leads to an effective and efficient knowledge representation. Therefore, as posed in Figure 1.2 we propose five research questions that are explained below.

RQ1 — What are the existing OIE methods and challenges faced to advance triple extraction and information retrieval from unstructured texts?

Early IE and IR systems emphasized answering well-defined requirements given a pre-defined set of target relations. This was mainly carried out using pattern-based approaches acquired from hand-labeling patterns. Manually

constructing patterns to cover a wide range of relations is deemed a tedious and unsustainable process with diverse relations across different domains. In 2007, OIE systems began to emerge and address the limitations associated with extracting pre-defined relations. Nevertheless, the extraction of information without relying on any existing rules comes with its drawbacks. Thus, several challenges arose in the OIE field mainly from corpus heterogeneity and the lack of labeled data for training such models. Different techniques to extract tuples are deployed in OIE, including learning-based, clause-based, and rule-based systems (Niklaus et al., 2018a). This research question investigates previously proposed OIE systems alongside the challenges associated with such strategies.

RQ2 — How can contextualized word embeddings enhance the process of Open Information Extraction?

Learning semantic representation for words has been an active research area in linguistics and NLP in the past decade. Conventional embedding approaches are primarily statistical and rule-based methods, attempting to model the distributional semantics. Later, pre-trained word embedding models were incorporated into many NLP tasks owing to their capability of abstracting semantic and syntactic features. These are usually loaded into the embedding layer of a NN, which converts tokens into vector space equivalents. Nevertheless, the impact of word embeddings differs from one model to the other on the performance of the NN model. To answer this research question, we begin by designing a novel bidirectional GRU-based OIE model and subsequently focus on assessing different word embeddings' effects on OIE.

RQ3 — How transferable is OIE to other NLP tasks to diminish the complication of insufficient training data of neural network models and aid in model generalization?

Transfer learning is considered as one of the most powerful deep learning technologies, as it addresses the ongoing challenge of the lack of labeled data by taking the knowledge from one task and applying it to a different task. The existing labeled datasets for several IE tasks is generally modest in size, which hinders the efficiency of the training process. Additionally, the manual labor to create appropriate labeled training data for such tasks are expensive and simply unfeasible. For this research question, we are concerned with two aspects regarding the transferability of OIE: transferring from one domain to the other — domain adaptation — and transferring to a correlated task, namely

RE. In both aspects, experiments in transductive transfer learning- and inductive transfer learning are carried out to better understand the transferability effect of OIE.

RQ4 — How can the robustness of pre-trained language models be enhanced to generalize for unseen patterns during inference on downstream tasks?

Several variations of pre-trained language models are currently dominating across different NLP tasks; nonetheless, robustness and generalization have always been a persistent concern in the fine-tuning procedure of pre-trained language models. This influenced researchers to direct their efforts toward enhancing the robustness and stabilization of the fine-tuning process. Two reasons behind the instability of the pre-trained language model are catastrophic forgetting and limited dataset size that is available for fine-tuning the model (Mosbach et al., 2021). Predicting the taxonomic relationship in a given sentence is advantageous in many NLP applications, such as common sense validation and taxonomy construction. The SemEval-2022 task PreTens (Brunato et al., 2022) has focused on formulating this task as a classification and regression task to evaluate the correctness of the taxonomic relationship that holds between two nouns in a sentence. A key challenge in this task is the inadequate amount of labeled data available, the testing data set was highly diverse in comparison to the provided training data; this is reflected in the number of unseen patterns during the inference phase. This research question, therefore, investigates several techniques to enhance the generalizability of the pre-trained language model for the task of taxonomy classification.

RQ5 — How can OIE be employed to populate knowledge graphs to facilitate querying and retrieval of data?

Knowledge graphs are becoming indispensable components to represent, preserve, and organize knowledge; however, they are still heavily dependent on humanly-curated structured text. To automatically construct a knowledge graph, IE techniques are applied to harvest relevant information from an unstructured corpus. As the vocabulary in several domains tends to be widespread in many areas, various challenges arise when building effective IE systems, irrespective of the domain of discourse. In particular, relationships that bind entities are usually complex and may suffer from semantic variations of the same concept. Consequently, knowledge graphs suffer from erroneous information compounded by the fact that the validity of the extracted triples is not guaranteed. Therefore, this final research question aims to investigate the

Table 1.1: Overview of datasets used in each of the individual chapters.

Chapter	RQ	Research Method	Dataset
2	RQ1	Literature review	n/a
3	RQ2	Computational Experiment	News corpora Wikinews
4	RQ3	Computational Experiment	News corpora Wikinews Drug-Drug interactions
5	RQ4	Computational Experiment	SemEval-2022 Dataset
6	RQ5	Computational Experiment Case study	Advanced persistent threat reports Microsoft security bulletins Cyber threat intelligence reports

instantiation of the OIE task for knowledge graph construction by designing and evaluating an end-to-end system.

1.5 Dissertation Outline

The research questions RQ1–RQ5 presented in Section 1.4 are investigated in Chapters 2–6 of this dissertation, with each research question corresponding to a single chapter. Each of these five chapters is written as a self-contained paper and published in proceedings of scientific conferences or scientific journals. Table 1.1 denotes the empirically addressed research questions, research methods, and dataset type utilized. The dissertation is structured as follows:

Chapter 1 — Introduction describes the motivation and objectives of this research, along with the various IE techniques relevant to this dissertation. We introduce the main overarching research question, five more specific research questions, and research methods that are used to address these questions. Finally, the dissertation outline in this section describes how the remainder of this dissertation is structured.

Chapter 2 — Uncovering Algorithmic Approaches in Open Information Extraction explores existing OIE methods to comprehend the underlying limitations of the previously proposed approaches. Precisely, we analyze the performance

of pattern-based approaches alongside approaches that rely on machine learning classifiers to perform triple extraction. This chapter presents a comprehensive review of previous OIE models that are based on the above-mentioned approaches, along with the advantages and limitations of these models. Additionally, we investigate future trends and research directions that employ neural methods.

Published as — Sarhan, I. and Spruit, M. (2018). *Uncovering algorithmic approaches in open information extraction: A literature review.* In 30th Benelux Conference on Artificial Intelligence, pages 223–234. Springer CSAI/JADS.

Chapter 3 — Contextualized Word Embeddings in Open Information Extraction. In order to develop a rigid OIE system, we start off by building an RNN-based OIE model, as RNNs are suitable for tasks that involve sequential data as it captures the context of the sentence. As we completely rely on unstructured textual data, the sentence input might be relatively long. As time steps increase, so do the multiplications in the final gradient calculation. In order to alleviate problems associated with traditional RNN networks, we subsequently opt to employ a GRU network that is less likely to suffer from vanishing gradient. Further, we utilize a bidirectional network to have a final hidden state that merges the input sentence from both directions. Additionally in this chapter, we exhaustively compare several word embedding models with the objective of achieving the best performance in triple extraction.

.

Published as — Sarhan, I. and Spruit, M. (2019). *Contextualized word embeddings in a neural open information extraction model.* In Natural Language Processing and Information Systems (NLDB), pages 359–367, Cham. Springer International Publishing.

Chapter 4 — Transfer Learning for Open Information Extraction applies transfer learning algorithms to OIE by utilizing the features learned to extract relation triples from unstructured text. We aim to address the underlying technical challenges concerning the limited accessibility of labeled data that is suitable for training IE systems. To achieve that, we conduct experiments in both transductive and inductive transfer learning. We first start by leveraging the pre-trained OIE model on a news dataset, and transfer it to a bio-medical dataset. We then follow by transferring to a semantically correlated task, RE,

to investigate the adaptability of OIE to a different yet semantically-related NLP task. We argue that OIE can indeed act as a reliable source task.

Published as — Sarhan, I. and Spruit, M. (2020). *Can we survive without labelled data in NLP? Transfer learning for open information extraction.* Applied Sciences, 10(17):5758.

Chapter 5 — Enhancing the Generalizability of Language Models describes our participation in the SemEval-2022 task 3 PreTens: Presupposed Taxonomies Evaluating Neural Network Semantics for English, French and Italian languages (Brunato et al., 2022). The first sub-task aims to recognize the taxonomic relationship that connects two nouns in a sentence and classify whether it is sensible or not. The second sub-task formulates the problem as a regression task with the aim of assigning a score based on a seven-point Likert scale. For the first sub-task, we rely on fine-tuning pre-trained language models for taxonomy classification, however, as a consequence of the limited size of training data available, pre-trained language models fail to generalize well on unseen data. Substantial experiments were carried out including multi-task learning, data-enriched fine-tuning, and multi-stage learning. The best results were achieved using a two-stage fine-tuning process on ELECTRA. To increase the diversity of the training data, we relied on data augmentation techniques. We further investigate the effect of combining data from data augmentation methods, namely: back-translation, insertion, and substitution techniques, to improve performance on the target dataset. As for the second sub-task, the provided training data had only four unique patterns, thus, fine-tuning the language models were proven to be ineffective. Therefore, to address sub-task two we propose a simple yet efficient model that trains a regressor using features obtained from sentence embeddings. Intensive experiments were carried out to single out the best-performing model for each of the three languages.

Published as — Sarhan, I., Mosteiro, P. and Spruit, M. (2022). *UU-Tax at SemEval-2022 Task 3: Improving the generalizability of language models for taxonomy classification through data augmentation* In The 16th International Workshop on Semantic Evaluation, NAACL.

Chapter 6 —End-to-end System for Knowledge Graph construction using Open Information Extraction proposes a system to construct a knowledge graph using the extracted relation triples from an OIE system. Currently, the majority

of IE systems either impose a pre-determined set of relations or utilize an existing ontology, which restricts both the amount and diversity of the extracted information. Subsequently, this hinders the coverage of the knowledge graph. This is the primary motivation behind Open-CyKG, to build a data structure for a deeper knowledge understanding that permits efficient and effective querying by mining information from unstructured text. As a case study, we utilize Cyber Threat Intelligence (CTI) data, which is composed of evidence-based knowledge including context mechanisms, indicators, implications, and actional advice, as this information will be highly beneficial for attack detection and mitigation. We start by designing an attention-based OIE model to extract relational tuples from malware reports. Once we have extracted the relational triples, NER is employed to populate the KG. Finally, to enhance the quality of the KG, we then perform canonicalization and fusion techniques to aid cybersecurity analysts in comprehending how the CTI information is related.

Published as — Sarhan, I. and Spruit, M. (2021). *Open-CyKg: An open cyber threat intelligence knowledge graph.* Knowledge-Based Systems, 233:107524

Chapter 7 — Conclusion provides answers to the research questions, based on the investigations in the preceeding chapters. Furthermore, we elaborate on the best practices-based guidelines for improving the reusability and transparency. Additionally, we describe the research outcomes of this dissertation, limitations and directions for further research, and close with personal reflections.

1.6 Thesis Outcomes

In this section, we list the code, datasets, and publications of this thesis.

Code

1. *Multi-stage fine-tuning method for taxonomy classification* — It involves a two-stage fine-tuning for the pre-trained language model, ELECTRA, to classify whether the taxonomic relation that binds two nominal arguments is valid or not. This code was employed in our proposed model UU-

Tax in the work described in Chapter 5. The code is available at `https://github.com/IS5882/UU-TAX/tree/main/Sub-task%201%20Code`.

2. *Sentence encoder for taxonomic relationship acceptability score* — This code was employed in our proposed model UU-Tax in the work described in Chapter 5. We utilize features from the Universal Sentence Encoder (USE) to train several regressors includes; Linear Regression (LR), K-Nearest Neighbors Regressor (KNR), Decision Tree (DT), and Support Vector Regressor (SVR). The code is available at `https://github.com/IS5882/UU-TAX/tree/main/Sub-task%202%20Code`.

3. *Taxonomy classification procedure using multi-task learning* — Is the code of the experiment described in Chapter 5 where we utilize a commonsense validation task to fine-tune the language model in the first stage. In the second stage of fine-tuning we use the main dataset of the downstream task. The code is available at `https://github.com/IS5882/UU-TAX/tree/main/Multi-task%20fine-tuning`.

4. *Data-enriched fine-tuning method for taxonomy classification* — Is the implementation of the experiment defined in Chapter 5 that employs an LSTM on top of BERT to assign an acceptability label of the taxonomic relationship between two nouns in a given sentence. In addition to the input sentence, we input to BERT the two nominal arguments which are extracted using TF-IDF. The code is available at `https://github.com/IS5882/UU-TAX/tree/main/Data-enriched%20fine-tuning`.

5. *Attention-based model for open information extraction* — Is the code implementation of the proposed encoder-decoder OIE model that employs a Bi-GRU followed by an attention component as described in Chapter 6 which is the main building block of the Open-CyKG methodology. The code is available at `https://github.com/IS5882/Open-CyKG/blob/main/Open_CyKG_OIE_Model.ipynb`.

6. *Named entity recognizer for cybersecurity terms* — Is the implementation of the second component of the proposed Open-CyKG methodology described in Chapter 6. The proposed NER is employed to label prominent cybersecurity terms. The code is available at `https://github.com/IS5882/Open-CyKG/blob/main/Open_Cy_KG_NER.ipynb`.

7. *Hierarchical agglomerative clustering process for knowledge graph canonicalization* — Is the software implementation of the canonicalization and fusion process of the KG in the Open-CyKG methodology as described

in Chapter 6. The code is available at `https://github.com/IS5882/Open-CyKG/blob/main/Open_CyKG__Knowledge_Graph_Canonicalization.ipynb`.

Datasets

1. *Augmented dataset for taxonomy classification task* — Is the augmented version of SemEval-2022 Task 3 PreTens described in Chapter 5. The augmentation is obtained from back-translation of the provided languages – English, French and Italian –. For each language l, we translate the datasets of the other two languages into l. In addition, the augmented data generated from the NLPAug tool employs BERT to substitute and insert words in each sentence. The back-translated dataset and the NLPAug dataset are available at `https://github.com/IS5882/UU-TAX/tree/main/Translation%20Data` and `https://github.com/IS5882/UU-TAX/tree/main/NLPAug%20Data` respectively.

2. *Cybersecurity named entities corpora* — Is the manually labeled portion of cybersecurity terms used to evaluate the performance of our proposed NER model in the Open-CyKG methodology described in Chapter 6. The dataset is available at `https://github.com/IS5882/Open-CyKG/tree/main/NER_CS`.

3. *Canonicalized knowledge graph* — Is the gold standard cluster used to evaluate the canonicalized knowledge graph generated from the Open-CyKG methodology as described in Chapter 6. The gold standard is available at `https://github.com/IS5882/Open-CyKG/tree/main/KG_goldStandard`.

Publications

Publications that are incorporated into this thesis:

1. Sarhan, I. and Spruit, M. (2018). *Uncovering algorithmic approaches in open information extraction: A literature review.* In 30th Benelux Conference on Artificial Intelligence, pages 223–234. Springer CSAI/JADS.

2. Sarhan, I. and Spruit, M. (2019). *Contextualized word embeddings in a neural open information extraction model.* In Natural Language Processing and Information Systems (NLDB), pages 359–367, Cham. Springer International Publishing.

3. Sarhan, I. and Spruit, M. (2020). *Can we survive without labelled data in NLP? Transfer learning for open information extraction.* Applied Sciences, 10(17):5758.

4. Sarhan, I. and Spruit, M. (2021). *Open-CyKg: An open cyber threat intelligence knowledge graph.* Knowledge-Based Systems, 233:107524

5. Sarhan, I., Mosteiro, P. and Spruit, M. (2022). *UU-Tax at SemEval-2022 Task 3: Improving the generalizability of language models for taxonomy classification through data augmentation* In The 16th International Workshop on Semantic Evaluation, NAACL.

A full list of publications can found on page 175.

2 Uncovering Algorithmic Approaches in Open Information Extraction: A literature Review

The explosion of mostly unstructured data has further motivated researchers to focus on Natural Language Processing (NLP), hereby encouraging the development of Information Extraction (IE) techniques that target the retrieval of crucial information from unstructured texts. In this paper we present a literature review on Open Information Extraction (OIE). We compare both machine learning and handcrafted rules-based algorithmic approaches and identify the recently proposed Neural OIE approach as a particularly promising area for further research.

This work was originally published as:
Sarhan, I. and Spruit, M. (2018). Uncovering algorithmic approaches in open information extraction: A literature review. In *30th Benelux Conference on Artificial Intelligence*, pages 223–234. Springer CSAI/JADS

2.1 Introduction

As the demand for a fast and efficient method to extract pivotal information from text increases day by day, researchers are encouraged toward IE tasks. OIE is the process of extracting relation tuples from text. It targets to ease the process of identifying domain-independent relations extracted from texts that scale to large-size data. OIE executes either a single or a consistent number of passes over its corpus, capturing vital relationships represented in each clause in the form of relational tuples (Christensen et al., 2011a).

The key difference between the Relation Extraction (RE) task and OIE is that OIE does not require a specific predefined relation domain, simply, the relation extracted is the text that links the two arguments together. In a domain-specific RE approach, the relation in interest should be pre-specified. For instance, given the sentence *"Barack Obama born August 4, 1961, in Hawaii served as the 44*th *President of the United States"*. A (BornIn-Loc) relation will extract the following arguments ≪ *Barack Obama,Hawaii* ≫. In contrast to OIE that will extract the following relation triples in the format ≪ *argument 1, relation, argument 2* ≫:

- ≪ Barack Obama, BornIn-Loc, Hawaii ≫
- ≪ Barack Obama, BornIn-Year, August 4, 1961 ≫
- ≪ Barack Obama, Served-as, President of the United States ≫

The extracted tuples can be binary, ternary, or n-ary, where the relationship is expressed between more than two entities, such as Person-Location-Organization relation ≪*John Smith, California, XYZ Company* ≫. OIE can be represented in two broad categories, approaches that require machine-generated data to train a classifier and approaches that rely on hand-crafted rules (Gamallo, 2014). Each category is further divided into two sub-categories as shown in Figure 2.1.

This chapter analyzes different OIE approaches and gives a glimpse into the future of OIE. The remainder of this paper is structured as follows; Section 2.2 describes the methods and search strategy, while Section 2.3 presents the first type of OIE that utilizes machine learning classifiers, followed by the second paradigm that is based on hand-crafted rules in Section 2.4. Section 2.5 briefly discusses OIE challenges. Section 2.6 explores the new trends in OIE. Finally, Section 2.7 concludes the literature review.

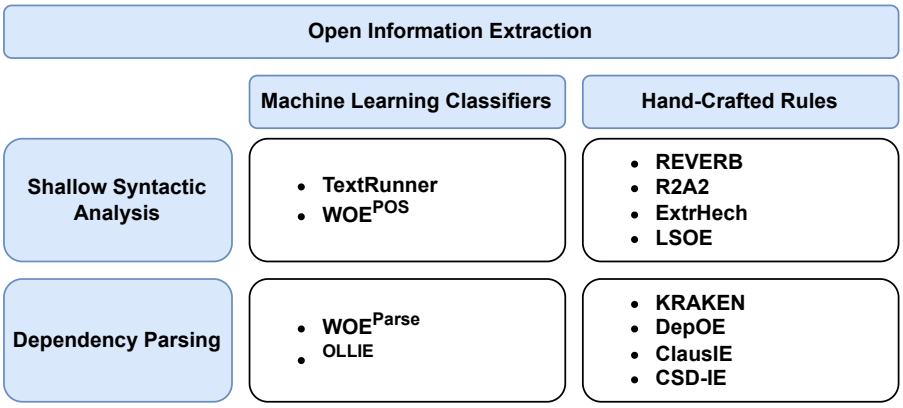

Figure 2.1: Open Information Extraction Categories.

2.2 Methods and Search Strategies

The structured search was carried out in May-June 2018. The Snowball method was employed for this literature survey and citation searching. After extensive research, we included papers that impacted the OIE field, with the example of TextRunner (Yates et al., 2007) as the initial OIE system. As the field of OIE has proliferated in the last decade, with the exception of Yates et al. (2007), only articles published in the last eight years are included in our survey paper.

2.3 Machine Learning Classifiers

In this section, we overview the OIE systems that we have studied that utilize automatically generated data to train a classifier. This methodology is further divided into two subcategories; those approaches that use shallow syntactic analysis and approaches that utilize dependency parsing.

Shallow Syntactic Analysis

TextRunner, the first OIE system, is a fully implemented, highly adaptable system introduced by Yates et al. (2007). TextRunner utilizes a Naïve Bayes model to determine if the heuristically selected tokens that lie between two

entities indicate a relationship or not. It exploits a domain-independent technique to extract relation tuples from a text corpus. Afterward, it identifies domain-specific terms in the tuples by utilizing class recognizers, thus learning relation mapping rules and finally transforming the tuples into domain relations (Soderland et al., 2010).

A corpus of 9 million web pages is the sole input to TextRunner, which then executes the extraction process in 3 key steps (Christensen et al., 2010):

1. A self-supervised learner: Low demand for hand-labeled training data is required due to the self-supervised nature of the system. The learner produces a Conditional Random Field (CRF) based classifier that exploits unlexicalized features in order to extract relations from the corpus.

2. A single pass extractor: The system extracts all possible relation tuples by making a single pass over the corpus using the CRF classifier. The extractor retains the tuples that are classified as trustworthy.

3. A redundancy-based assessor: The extracted tuples are re-ranked based on a probabilistic redundancy model — similar to the one used in KnowItAll (Etzioni et al., 2005) —. The assessor allocates a confidence score to each extracted tuple based on its number of occurrences in the text.

All these components enable TextRunner to be a high-performance, general, and high-quality extractor for heterogeneous web text. Subsequent work showed that utilizing a linear-chain CRF (Banko and Etzioni, 2008) or Markov Logic Network (Zhu et al., 2009) leads to further improvements over TextRunner (Etzioni et al., 2011).

Wu and Weld introduced the WOE (Wikipedia-based Open Extractor) system (Wu and Weld, 2010) that can operate in two modes: WOE^{POS} and WOE^{Parse}. The WOE^{POS} approach uses a CRF extractor trained with shallow syntactic features, unlike WOE^{Parse}, which is discussed in the next section, WOE^{POS} system enhances TextRunner's performance by utilizing Wikipedia to train data for their extractors. The primary concept behind WOE is the automatic assembly of training examples by heuristically matching Wikipedia info box values with corresponding text. When compared to TextRunner, WOE^{POS} increases the F-Measure by almost 34% owing to finer training data from Wikipedia via self-supervision.

Dependency Parsing

Wu and Weld (Wu and Weld, 2010) additionally demonstrated that dependency parse features cause a massive increase in both recall and precision when

compared to shallow linguistic features. Thus they introduced the aforementioned model WOE^{Parse}. The parser-based extractor — WOE^{Parse} — utilizes a plentiful dictionary of dependency path patterns acquired from heuristic extractions produced from Wikipedia. WOE^{Parse} reaches an F-measure between 72% and 91% higher than that of TextRunner. The main reason behind this increase is because complex sentences with complicated distance relations are handled better using a parser, and this results in WOE^{Parse} to maintain a decent recall with only tolerable loss of precision, it also outperforms WOE^{POS}. Albeit, it runs around 30 times slower than TextRunner owing to the time required for parsing.

OLLIE (Open Language Learning for Information Extraction) system, introduced by Mausam et al. (2012), overcomes the main drawbacks of ReVerb (Fader et al., 2011) — discussed in Section 2.4 — and WOE, where extraction of both non-factual tuples and relations are only intervened by verbs. OLLIE bootstraps an immense training set from a number of high-precision seed tuples obtained from ReVerb to learn semi-lexicalized pattern templates. Those pattern templates are features in a dependency parse as they determine both the argument and the relation phrase. They are later put into use during the extraction phase (Mausam, 2016). The concept behind the learning component is to retrieve a large number of example sentences that assert a specific tuple to ensure that all the essential information has been captured. Eventually, OLLIE investigates the text around the tuple to append more details (attribution and clausal modifiers). The authors of OLLIE signal out certain features that seem to capture nearly all of the sentences with attribution and clausal modification. For both attribution and clausal, a feature and a filter are needed to remove false positives. Finally, OLLIE's confidence function is trained to lessen the confidence of an extraction if its surrounding text indicates that there is a possibility that it is non-factual. OLLIE achieved an area under the curve (AUC) of 2.7 times higher than ReVerb and 1.9 times larger than WOE^{Parse}.

2.4 Based on Hand-crafted Rules

The second category of OIE methods makes use of hand-crafted rules or heuristics to extract relation triples. Similar to the first category of unsupervised training of classifiers, it is also further divided into two subcategories; those approaches that use shallow syntactic analysis and approaches that utilize dependency parsing.

Shallow Syntactic Analysis

In 2011, Fader et al. (2011) proposed ReVerb, which resembles the TextRunner approach by utilizing computationally efficient surface patterns over tokens. ReVerb implements a general model of verb-based relation phrases by applying two simple constraints to extract binary facts. This algorithm consists of three major steps:

1. Identify relation phrases that meet syntactic and lexical constraints.

2. Locates a pair of Noun Phrase (NP) arguments for each identified relation phrase.

3. A confidence score is then allocated to the resulting extractions using a logistic regression classifier trained on 1,000 random Web sentences with shallow syntactic features.

In other words, ReVerb takes as input a POS-tagged and NP-chunked sentence and outputs a set of $\ll$ *NP1, relation, NP2* $\gg$ extraction triples. The inclusion of syntactic constraints aided in reducing uninformative extractions. Furthermore, a lexical constraint separates valid relation phrases from over-specified relation phrases. This algorithm deviates from the TextRunner algorithm in four significant manners: First, the relation phrase is recognized "holistically" rather than word-by-word as in TextRunner. Second, possible relation phrases are filtered based on statistics over a sizable corpus. Third, the ReVerb algorithm works on extracting the relation first, and then it extracts the argument, as a result of this avoiding the confusion between a noun in the relation phrase for an argument and finally, by introducing lexical and syntactic constraints resulting in doubling the area under the precision-recall curve when compared to TextRunner and WOE^{POS}.

Succeeding the aforementioned approaches, Etzioni et al. (2011) introduced the second generation of OIE, R2A2, through merging ReVerb with an argument identifier — ArgLearner — to enhance argument extraction for the relation phrases. ArgLearner is a learning-based system. When given a sentence and a relation phrase pair, it identifies arguments by utilizing patterns as features. In both TextRunner and REVERB, the arguments are the two adjacent NPs, while R2A2 utilizes ArgLearner to learn independent extractors for the left and right boundaries of each argument using three classifiers, two of which identify the left and right bounds of ARG1 and the third classifier identifies the right bound of ARG2 (Etzioni et al., 2011). The system is then compared against ReVerb on two datasets, each consisting of 200 random sen-

tences. R2A2 increased the area under the precision-recall curve by almost 100% from 0.45 to 0.9 (Etzioni et al., 2011).

Xavier et al. (2013) debate that it is not compulsory to have an immense list of patterns or various kinds of linguistic labels to perform OIE. In order to prove their proclaimed theory, they developed LSOE (Lexical Syntactic pattern based Open Extractor), a novel unsupervised OIE approach that implements lexical-syntactic patterns to POS-tagged texts to extract relation triples $\ll$ *Arg1, Relation, Arg2* $\gg$. The strategy is based on two types of patterns:

1. Generic patterns to identify non-specific relations.

2. Rule-based patterns to learn Qualia structure.

LSOE performance was compared with two state-of-the-art Open IE systems: ReVerb and DepOE (Gamallo et al., 2012). The latter is discussed in the next section. LSOE attained a higher precision when compared to the aforementioned state-of-art approaches.

Another approach that exploits shallow syntactic analysis based on hand-crafted rules is ExtrHech (Zhila and Gelbukh, 2013). The latter approach acquires Part of Speech (POS) tagged text as input, applies syntactic constraints as regular expressions, and outputs a set of relation triples. It is worth noting that ExtrHech is a multilingual system that's applied in Spanish and English. When compared against ReVerb on a 68-sentence dataset, ExtrHech outperformed ReVerb in terms of precision and recall.

Dependency Parsing

KRAKEN (Akbik and Löser, 2012), an OIE methodology particularly purposed to capture complete N-ary facts, in addition, to examining fact completeness and correctness, which reflects on the quality of the extracted data. Given a Stanford-parsed sentence as an input to the system, KRAKEN conducts the following three main steps:

1. Fact phrase detection: KRAKEN locates fact phrases as a series of verbs, modifiers, or prepositions.

2. Detection of argument heads: Using type-paths, heads of arguments can be identified. Every type-path indicates one or more links and the direction of each link to follow to find an argument head.

3. Detection of full argument: Recursively trail all downward links from the argument head to get the full argument.

Hand-crafted rules are used to locate relational phrases, and their corresponding argument over typed dependency parses. Provided that a fact phrase has at least one argument, the system extracts it as a fact. When KRAKEN is compared against ReVerb, KRAKEN almost doubles the number of recognized complete and actual facts. It achieves notable results for binary, ternary, and 4-ary facts.

DepOE (Dependency-Based Open Information Extraction) (Gamallo et al., 2012) is a multilingual OIE system specifically designed to extract verb-based triples from Wikipedia in four languages: Portuguese, Spanish, Galician, and English. The latter system embraced the features of both Machine Reading by creating an efficient and fast system guaranteeing scalability as the corpus grows and Learning by Reading by utilizing a dependency-based parser beneficial to obtaining fine-grained information. DepOE relies on the following steps:

1. Dependency parsing: All the sentences of the input text are inspected by the dependency- based parser by a multilingual tool.

2. Clause constituents: For each parsed sentence, it discovers the verb clauses it contains and, then, for each clause, it locates the verb candidates [subject, direct object, attribute, and prepositional complements].

3. Extraction rules: A group of rules are employed on the clause constituents that are extracted from the previous step to extract the target triples.

Nevertheless, the dependency-based parser that was used in the first step has insufficient grammar that results in partial parsing lacking deep analysis. As a result, the number of extracted triples by DepOE is fewer than ReVerb. It is worth noting that DepOE has more accurate extractions of the two arguments as opposed to ReVerb, that suffers from the erroneous identification of the first argument and extraction of an incomplete part of the second argument.

In the following year, Del Corro and Gemulla introduced ClausIE (Clause-based Open Information Extraction) (Del Corro and Gemulla, 2013). ClausIE benefits from the linguistic knowledge of the grammar of the English language to identify clauses in an input sentence and afterward determines the category of each clause to be consistent with the grammatical function of its constituents. Given an input sentence, ClausIE performs the following:

1. Computes the dependency parse of the input sentence to discover its syntactical structure.
2. Specify the set of clauses using the dependency parse.
3. Learn the set of coherent derived clauses based on the dependency parse and small domain-independent lexica.
4. Generate one or more propositions for each clause.

ClausIE primarily differs from preceding systems in the way that it does not exploit any training data and also does not necessitate further processing to remove low-precision extractions, in contrast to ReVerb and OLLIE, both use post-processing statistical techniques that aid in increasing the precision. ClausIE yields 2.5–3.5 times more correct extractions than OLLIE. The inclusion of low-confidence propositions declines precision, which explains why TextRunner's precision is significantly lower than that of ReVerb, WOE, and ClausIE. The latter three extractors obtain high precision due to high-confidence propositions.

Bast and Haussmann demonstrate that contextual sentence decomposition, a method initially created for high-precision semantic search, can also be utilized for OIE, hereby introducing CSD-IE (Contextual Sentence Decomposition Information Extraction) (Bast and Haussmann, 2013). CSD-IE is carried out in two primary steps:

1. Identification of fundamental building blocks of the desired contexts in the sentence constituent identification (SCI) phase. Thus, a tree expressing the semantics is derived.
2. Tree constituents are combined to form the contexts creating sentence constituent recombination (SCR).

Triples are then obtained from the context's outcome by identifying the first explicit verb phrase and surrounding adverbs to be the predicate, with the prefix being the subject and the postfix being the object. This approach achieves a decent precision with high recall and very good coverage and minimality when compared against ReVerb, OLLIE and ClauseIE.

Despite the fact that WOE and OLLIE both exploit dependency parsers, they yet fail to determine the subject of the second clause correctly, owing to their use of automatically learned dependency parser patterns, see Table 2.1. For instance, the OLLIE system learns from ReVerb. Another consequence of using automatically learned dependency parser patterns is the high number

Table 2.1: Comparison of WOE and OLLIE extractions with true extractions, highlighting incorrect identification of the subject of the second clause. (Adapted from Del Corro and Gemulla (2013))

Sentence	The principal opposition parties boycotted the polls after accusations of vote-rigging, and the only other name on the ballot was a little-known challenger from a marginal political party.
WOE	≪ the only other name, was, a little-known challenger ≫
OLLIE	≪ the only other name, was, a little-known challenger ≫
True triples	≪ The principal opposition parties, boycotted, the polls ≫ ≪ The principal opposition parties, boycotted, the polls after accusations of vote-rigging ≫ ≪ the only other name on the ballot, was, a little-known challenger ≫ ≪ the only other name on the ballot, was, a little-known challenger from a marginal political party ≫

of incorrect extractions produced by OLLIE. Hand-crafted approaches using dependency parsing, therefore, seem the way to go. However, these approaches still suffer from error propagation caused by the employed patterns.

2.5 OIE Challenges

Extracting data from the text might be challenging. The two most recurrent challenges that many of the aforementioned OIE approaches face are uninformative and incoherent extractions (Fader et al., 2011).

Incorrect handling of relational phrases is the main root of uninformative extraction that results in leaving out crucial information. For further illustration, consider the following sentence *"John Smith signed a fixed-price contract with ABC company after a 2-month negotiation period"*, uninformative extraction results in extracting ≪ *John Smith, signed, fixed-price contract* ≫ instead of extracting ≪ *John Smith, signed a contract, ABC company* ≫. Uninformative extractions make up almost 4% of WOE^{Parse} output, 6% of WOE^{POS} output, and 7% of TextRunner's output (Etzioni et al., 2011).

Incoherent extraction is purposeless extractions derived from opaque relation phrases that the extractor fails to correctly identify, as is the case with major OIE state-of-art (Yates et al., 2007; Wu and Weld, 2010). Considering the previous sentence, an example of incoherent extraction would be ≪ *John Smith, signed a contract, 2-month negotiation period* ≫. Incoherent extractions form nearly 30% of WOE^{Parse} output, 15% of WOE^{POS}, and 13% of

TextRunner's output (Etzioni et al., 2011). Syntactic and lexical constraints reduce uninformative extractions and exclude incoherent extractions, in addition to decreasing overly-specified extraction.

,ReVerb exploits a syntactic constraint to overcome this constraint that forces every multi-word relation phrase to start with a verb, ending with a preposition, and be a neighboring sequence of words in the sentence. Hence, it prevents the extractor from making a series of decisions to decide whether to include each word in the relation phrase or not, regularly resulting in unclear predictions.

Most of the current OIE approaches center their research on the extraction of binary facts and suffer a notable quality deterioration when capturing higher order N-ary relations, except for KRAKEN, which focuses on the extraction of N-ary facts.

2.6 Future Trends in OIE

Recently, Neural Networks (NN) methods have been gaining massive attention due to their proven success at tackling various NLP tasks (Rush et al., 2015; Gehring et al., 2017; Meng et al., 2017). Distinctively from the several OIE state-of-the-art systems discussed in this paper, Cui et al. (2018) proposed a neural OIE paradigm that implements an encoder-decoder framework. The encoder-decoder infrastructure is a method for text generation and has already been utilized in other NLP tasks successfully (Cui et al., 2018). Being implemented by a recurrent neural network, the encoder-decoder framework inputs a variable-length sequence, then the decoder uses the resulting compressed representation vector to produce the output sequence. The encoder and decoder use a 3-layer Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997). Binary extractions with high confidence are used to train the proposed neural OIE approach bootstrapped from a state-of-the-art OIE system, resulting in the generation of high-quality tuples.

While several OIE approaches have been developed in the past decade to extract relations from given corpora, mainly in the English language, only a few researchers target other languages (Xavier et al., 2013; Gamallo et al., 2012). Future research should be aimed toward developing a multi-lingual OIE paradigm.

Furthermore, as previously discussed in the previous section, until now, the main focus has been on the extraction of binary relations, omitting the importance of extraction of higher order relations that has a high impact not only on the quality of the extracted relations but also its completeness and

Table 2.2: Comparison of the different OIE systems discussed in this chapter.

Approach	Category (Sub-category)	Dataset	Advantages	Disadvantages
TextRunner	Machine Learning Classifier (Shallow Syntactic Analysis)	9 Million Webpages	First OIE system with a single-pass over corpus	Low precision caused by the addition of low confidence propositions
WOE^{POS}	Machine Learning Classifier (Shallow Syntactic Analysis)	Penn Treebank, Wikipedia, and the general Web	Unlike WOE^{Parse}, it avoids extracting false high- confidence triples.	Does not use deep processing during extractions.
WOE^{Parse}	Machine Learning Classifier (Dependency Parsing)	Penn Treebank, Wikipedia, and the general Web.	Proved that parsed features play an important role in informative extractions	WOE^{Parse} processes each sentence with a dependency parser, thus requiring in a longer processing time
OLLIE	Machine Learning Classifier (Dependency Parsing)	300 sentences from 3 sources: News, Wikipedia and Biology textbook	Applies deep syntactic analysis to extract new relations	Significantly slower than other state-of-art OIE systems
ReVerb	Hand-Crafted Rules (Shallow Syntactic Analysis)	Same dataset as TextRunner trained on Penn Treebank	Employing syntactic constraint to avoid uninformative extractions	REVERB restricts subjects to noun phrases without prepositions
R2A2	Hand-Crafted Rules (Shallow Syntactic Analysis)	20,000 sentences	Utilize ArgLearner to better Identify the arguments	Does not identify long tail of patterns thus misses important recall from verb-based relations with long range dependencies
LSOE	Hand-Crafted Rules (Shallow Syntactic Analysis)	9 million Web documents.	Identifying non-specific relations using generic patterns.	The lack of lexical syntactic patterns results in missing any relations expressed by verbs
ExtrHech	Hand-Crafted Rules (Shallow Syntactic Analysis)	68 sentences from FactSpaCIC	Multi-lingual OIE system (English and Spanish)	System is evaluated on a small corpus
KRAKEN	Hand-Crafted Rules (Dependency Parsing)	500 sentences sampled from the Web using Yahoo's random link service	Extraction of n-ary facts	More than quarter of the evaluation set was skipped since the dependency parser employed does nott indicate uncertain grammatical relationships
DepOE	Hand-Crafted Rules (Dependency Parsing)	Sports and Biology articles from Wikipedia	Supports Multi-lingual extractions (English, Spanish, Portuguese, and Galician) by utilizing a multilingual rule-based parser	Unable to correctly extract arguments in the English version due to the inefficiency of the named entity recognition system
ClausIE	Hand-Crafted Rules (Dependency Parsing)	3 datasets: ReVerb dataset, 200 random sentences from Wikipedia, 200 random sentence form NY Times.	Extracts non-verb mediated propositions.	Incorrect dependency parses and implementation tends to miss essential adverbials.
CSD-IE	Hand-Crafted Rules (Dependency Parsing)	2 datasets from ClausIE: 200 random sentences from the Wikipedia, and 200 random sentences from the NY Times	Achieves minimality, to increase the relevance extracted arguments and relation by further decreasing its size	Errors due to incorrect parsing
Neural OIE	Neural Network	Benchmark dataset from Stanovsky et al. (2018) that contains 3,200 sentences	Avoiding error propagation caused by hand-crafted pattern by employing an encoder-decoder framework	Only supports binary extractions and non-nested relations

correctness.

2.7 Conclusion

With the ongoing advancements in the field of NLP, OIE has become increasingly popular in the past years. Practically, the current OIE paradigms either employ automatically assembled training data or hand-crafted heuristics.

We started by reviewing approaches that necessitate machine learning classifiers. TextRunner (Yates et al., 2007) and WOE^{POS} (Wu and Weld, 2010) emphasize improving the efficiency of the extracted triples by applying syntactic constraints as POS and chunking. WOE^{Parse} (Wu and Weld, 2010) and OLLIE (Mausam et al., 2012) use dependency parse features to boost the recall and precision, even though this affects negatively on the extraction speed. OLLIE stood out by achieving a higher AUC when compared to ReVerb and WOE^{Parse}.

ReVerb (Fader et al., 2011) used Hand-crafted patterns and exploited lexical and syntactic constraints to extract relation triples, achieving notable results. While R2A2 (Etzioni et al., 2011) further enhanced ReVerb by employing an argument learner. The second type of hand-crafted rules relied on dependency parsing like ClausIE (Del Corro and Gemulla, 2013), KRAKEN (Akbik and Löser, 2012) and DepOE (Gamallo et al., 2012). A summary of all the discussed approaches can be found in Table 2.2.

The analysis of this survey appears to primarily support the second approach, using hand-crafted patterns as is shown in the evaluation of (Akbik and Löser, 2012; Del Corro and Gemulla, 2013). However, after reviewing OIE systems, we believe that future research should be more directed toward a Neural Network (NN) approach. NN has already provided a boost to several NLP tasks. The model proposed by (Cui et al., 2018) overcame the error propagation caused by hand-crafted rules.

In conclusion, there is still room for improvement in OIE. OIE cannot be regarded as a simple NLP task, and it still faces several shortcomings that open up many research questions. While we have tried to cover the most representative state-of-art approaches that have appeared in the modern literature to get a complete picture, (Niklaus et al., 2018a) also recently reviewed and assessed the performance of several OIE approaches. Their findings align with our work and underline the rapidly increasing interest in the quickly evolving OIE domain.

3 Contextualized Word Embeddings in a Neural Open Information Extraction Model

Open Information Extraction (OIE) is a challenging task of extracting relation tuples from an unstructured corpus. While several OIE algorithms have been developed in the past decade, only a few employ deep learning techniques. This study presents a novel OIE neural model that leverages Recurrent Neural Networks (RNNs) using Gated Recurrent Units (GRUs). Moreover, we integrate innovative contextual word embeddings into our OIE model, which further enhances the performance. The results demonstrate that our proposed neural OIE model outperforms the existing state-of-art on two datasets.

This work was originally published as:
Sarhan, I. and Spruit, M. R. (2019). Contextualized word embeddings in a neural open information extraction model. In *Natural Language Processing and Information Systems*, pages 359–367, Cham. Springer International Publishing

3.1 Introduction

Natural Language Processing (NLP) techniques that facilitate the process of fetching important information from large data are highly demanded. With the ongoing development in the field of NLP, OIE gained massive attention in the past years. OIE is the process of extracting a relation tuple from a text corpus in the form of ≪ *Entity1 - Relation - Entity 2* ≫ as seen in Table 3.1. It plays a fundamental role in turning massive, unstructured text corpora into factual information, and it can be used as a foundation for many NLP tasks, including information extraction, question answering, and summarization.

Table 3.1: Open information extraction example

Sentence	*Barack Obama born August 4, 1961 in Hawaii served as the 44th President*
Extracted tuples	≪ Barack Obama – Born - August 4 ≫ ≪ Barack Obama – Born - Hawaii ≫ ≪ Barack Obama – Served as- President of the USA ≫

Previously, OIE paradigms either utilized automatically assembled training data or hand-crafted heuristics. Nonetheless, after deep learning techniques paved their way in various NLP tasks, researchers aimed their focus toward neural networks. Recurrent Neural Network (RNN) is a robust class of artificial neural networks. Contrary to Feed-Forward networks, RRNs can loop among nodes. Thus it is capable of apprehending temporal behavior. This results in permitting information to persist in them by selecting which information to keep and which to forget by taking into consideration the current input and the previous data it received.

This study presents an OIE model that employs RNNs to extract relation triples. Recently, RNNs proved their importance by achieving notable performance in various NLP tasks such as translation (Cho et al., 2014) and speech recognition (Graves et al., 2013). They are heavily applied in Google Home (Li et al., 2017a) and Amazon's Alexa (Chung et al., 2017). The features that make RNNs a good fit for NLP applications are notable (Yin et al., 2017). For instance, they take into consideration the order of the words. In addition, GPUs can be utilized to carry out RNN computation. Therefore, they perform well on large datasets. Also, RNNs can handle arbitrary input and output lengths. Furthermore, we demonstrate that contextual embedding enhances the overall performance of the OIE task compared to non-contextual word embedding techniques.

The remainder of this chapter is structured as follows; Section 3.2 reviews the existing OIE state-of-art models, while Section 3.3 presents the proposed OIE model, followed by the results and evaluation in Section 3.4 Finally, Section 3.5 concludes the chapter and discusses future work.

3.2 Related Work

In this section, we review existing OIE state-of-art architectures. A complete picture can be found in the previous chapter — published as (Sarhan and Spruit, 2018) —. OIE can be categorized into two broad categories, approaches that require automatic machine learning classifiers and approaches that utilize handcrafted rules (Gamallo, 2014). Newly, deep learning techniques started paving their way towards OIE systems.

Machine Learning Classifiers

In 2007, Etzioni et al. (2008) introduced TextRunner, the first OIE system is a fully implemented, highly adaptable, self-supervised system that relies on shallow syntactic analysis. It makes use of a domain-independent technique on a text corpus in order to extract relation tuples. TextRunner extracts all possible relation tuples by making a single pass over the corpus using a Conditional Random Field (CRF) classifier. The extractor reserves tuples that are classified as trustworthy.

Wikipedia-based Open Extractor (WOE) system (Wu and Weld, 2010), introduced by Wu and Weld, operates in two modes: WOE^{POS} and WOE^{Parse}. The WOE^{POS} system employs a CRF extractor trained with shallow syntactic features, in contrast to WOE^{Parse}, which makes use of a rich dictionary of dependency path patterns. Heuristically matching Wikipedia info box values with corresponding text for automatic assembly of training examples is the primary idea behind WOE herby enhancing TextRunner's performance.

Hand-Crafted Rules

ReVerb proposed by Fader et al. (2011). ReVerb relies on the process of relation phrases that meet syntactic and lexical constraints. Afterward, it extracts noun phrase argument pairs for each relation phrase. A logistic regression classifier is later used to assign a confidence score for each extracted tuple. Subsequently, Etzioni et al. (2011) presented the second generation of

OIE, R2A2, by combing ReVerb with an argument identifier - ArgLearner - to enrich argument extraction for the relation phrases.

Del Corro and Gemulla proposed ClausIE, a clause-based OIE system that exploits the linguistic knowledge of the grammar of the English language to locate clauses in an input corpus (Del Corro and Gemulla, 2013). It determines the input sentence's dependency parse to realize its syntactic structure. Then, the algorithm acquires a set of coherent derived clauses based on the dependency parse and small domain-independent lexica and generates one or more propositions for each clause. ClausIE fundamentally varies from the aforementioned OIE systems in the way that it does not utilize any training data in contrast to ReVerb (Fader et al., 2011) and TextRunner (Etzioni et al., 2008).

Neural Approaches

A neural OIE paradigm was proposed by Cui et al. (2018) that employs an RNN encoder-decoder framework. The encoder-decoder infrastructure is a method for text generation and has already been utilized in other NLP tasks successfully, as illustrated in (Cui et al., 2018). The encoder inputs a variable-length sequence and outputs a compressed representation vector, which is then passed to the decoder, resulting in the output sequence produced by the decoder. Both the encoder and decoder use a 3-layer Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997). Training data is obtained from high-confidence binary extractions from a state-of-the-art OIE system. Thus, the extraction of high-quality tuples.

In addition to the work of Cui et al. (2018), authors in (Stanovsky et al., 2018) developed a Bidirectional LSTM transducer to extract OIE tuples, proving that supervised learning can have a substantial impact on OIE performance. By extending the work made on deep semantic role labeling to extract OIE tuples, authors of (Stanovsky et al., 2018) were able to achieve notable results. Moreover, their work emphasizes that research on Question Answering-Semantic Role Labeling paradigms can significantly benefit future OIE models.

3.3 Methods

Our proposed model is built on the work of (Stanovsky et al., 2018) by treating OIE task as a sequencing labeling problem resulting in the extraction of multiple, overlapping tuples for each sentence.

The proposed neural network framework takes a fixed-length vector of an embedded sentence as an input. In addition, predicates are the building blocks of any language. They denote decisive actions which are considered highly effective in extracting relations of interest. Thus, following the work of (Stanovsky et al., 2018), we assume that each sentence's predicate represents the relation associated with the tuple. Therefore the predicate is sent to the network as a feature vector along with the Part of Speech (POS) tag of the sentence using NLTK (Bird, 2006).

Contextual Embeddings

ELMo (Embedding from Language Models) (Peters et al., 2018) is a deep contextualized word representation that models both: complex syntactic and semantic features of a word and the way in which these words' uses differ throughout linguistics. The key idea behind ELMo is contextual embedding. Thus the representation of each word differs according to its neighboring words. The generated word vectors are acquired from the functions of the internal states of a deep bidirectional language model, which is pre-trained on a large dataset. We integrated ELMo embedding into our OIE model, and the results proved that contextual embedding yields better performance. The aforementioned neural OIE methods utilized either GloVe (Pennington et al., 2014) or Word2Vec (Mikolov et al., 2013b), both are non-contextual word embeddings. Comparative results are demonstrated in the subsequent section.

GRU Model Architecture

RNNs are hard to train due to the vanishing and the exploding gradient descent problems during the back-propagating process (Pascanu et al., 2013). Efforts were made to overcome this complication. Hence LSTMs and GRUs were developed. They both successfully dealt with the difficulty of training RNNs. Indeed, LSTM and GRU are considered very effective models for learning very long contexts. The way they are used in (Vukotic et al., 2016) allows training on long word contexts.

GRUs are comparatively new and employ fewer parameters than LSTMs, which eventually entails that GRUs are both lighter and faster to train than LSTMs. GRU merges LSTM's Input and Forget gate in the Update gate. In Addition, it merges the cell state and the hidden state, lowering the model's complexity. Contrary to LSTM, GRU has two gates instead of 3:

1. *Reset gate*: that decides how to integrate the previous memory with the

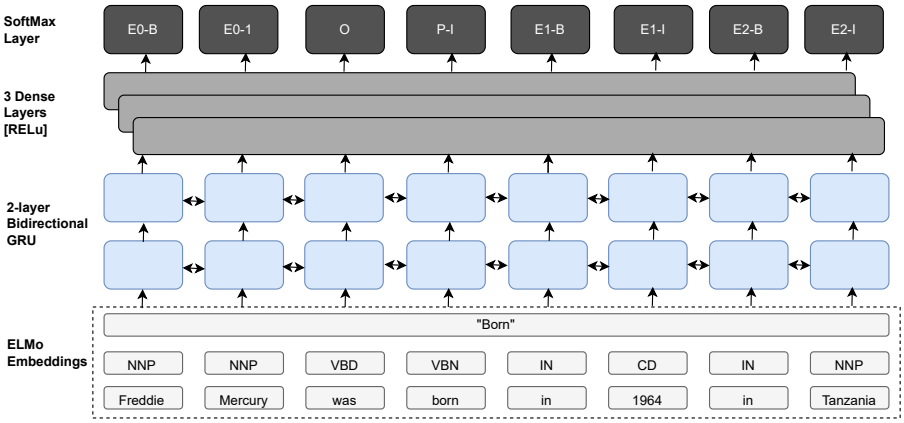

Figure 3.1: The proposed OIE model architecture for triple extraction.

current input.

2. *Update gate*: that determines the amount that it should keep from the prior memory. For GRU, the hidden state Ht is computed as defined by (Cho et al., 2014) in the equations below:

$$Z_t = \sigma\left(X_t \cdot U_z + H_t - W_z\right) \tag{3.1}$$

$$R_t = \sigma\left(X_t \cdot U_r + H_t - W_r\right) \tag{3.2}$$

$$h_t = tanh\left(X_t \cdot U_h + (R_t \cdot H_t - 1) \cdot W_h\right) \tag{3.3}$$

$$H_t = (1 - Z_t) \cdot h_t - 1 + Z_t \cdot h_t \tag{3.4}$$

Where Z and R denote the update gate and the reset gate, respectively, X represents the input vector, while U and W represent parameter vectors.

Our proposed OIE architecture is shown in Figure 3.1. In our OIE model, we implemented a 2-Layer Bidirectional GRUs. The default application of RNNs is to assess information in a single direction. However, it has been shown that modeling information in bidirectional technique results in better performance (Vukotic et al., 2016; Bansal et al., 2016). A Bidirectional GRU was employed to encapsulate the forward and backward lexical semantics of

each word in a given sentence. A bidirectional network can be generated in two different approaches, either by having two RNNs operating in opposing directions or within the internal architecture of the RNN itself. In our model, we employed the latter approach.

After encoding the three inputs using ELMo — the word, the POS tag of each word, and the predicate as shown in Equation 3.5 — they are all concatenated and passed as a single feature vector to the Bidirectional GRU. Subsequently, the Bidirectional GRU outputs a tensor that is passed to the three-layer Time Distributed Dense layer, which is later passed to the SoftMax layer for label prediction.

$$FeatureVector = ELMo(Word) \bigoplus ELMo(POS) \bigoplus ELMo(Predicate) \tag{3.5}$$

Eventually, the SoftMax layer assigns a probability of each word belonging to a particular label. We used BIO tags (Begin – Inside – Outside) (Ramshaw and Marcus, 1999a) that demonstrates the location of each word in the sentence, and each label is later assigned accordingly, as shown in the last layer in Figure 3.1. A sentence might include more than one entity, and each sentence may output more than one tuple, as the example in Table 3.1; however, our model captures binary relations. If a sentence contains no relation between the words, only the predicate is assigned as *"P-B"*, and label *"O"* is allocated to the remaining words in the sentence. Our work adopts the same approach proposed by Stanovsky et al. (2018), through presenting the proposition as a tuple that is comprised of a single predicate and a non-empty set of arguments. Both the predicate and arguments are continuous spans of text from the sentence. The order of the elements in the extracted triplets is structured to ensure natural interpretation when reading the tuple from left to right. This follows the principles of traditional binary OIE as identified in the works of Stanovsky et al. (2018), where each tuple must be supported by the sentence, and thus only extracting explicit relations, as explained in Section 1.2. Additionally, a sentence may contain multiple tuples, which we group by predicate head-word, identified using a POS tagger. As the predicate head-word is often the main verb and conveys the central idea of the sentence, grouping tuples this way enables us to run the model once for each predicate head and combine the results to produce the final set of extractions.

Table 3.2: Statistics of the datasets used in our experiments include the number of sentences in each dataset (#Sent) and the number of tuples (#Tuples) that are used during training, validation, and testing.

Dataset		Train set	Validation set	Test Set
Newswire	#Sent	744	249	248
	#Tuples	2173	727	737
WikiNews	#Sent	1174	392	393
	#Tuples	2906	946	993

Hyperparameters Settings

Our neural OIE architecture was implemented using Keras framework (Chollet, 2015) with TensorFlow backend (Abadi et al., 2016). Our model was trained on ten epochs with the dropout rate set to 0.1 for regularization to avoid over-fitting. The data is divided into 100 batches. Moreover, we use early stopping to terminate training when the performance stops improving. Each Bidirectional GRU has 128 units, which is the same number of the hidden units in the subsequent three Time Distributed Dense Layers (TDDLs). The activation function used in the three TDDLs is Rectified Linear Unit (ReLU) (Nair and Hinton, 2010a). Adam optimizer (Kingma and Ba, 2015) was employed to train our model.

3.4 Results and Evaluation

The performance of the proposed OIE model was tested on two different datasets. Three experiments were carried out to measure and compare the performance of the proposed BiGRU-based OIE approach using contextual embedding.

Dataset

The dataset we obtained for our model is further divided into two sets: Newswire corpus and Wikipedia News Corpus (WikiNews) (Stanovsky and Dagan, 2016). Our dataset is split into a training set to train the model, a validation set to validate the model, and a test set that is used to calculate the performance of our OIE proposed architecture. The number of sentences and number of tuples in each dataset can be found in Table 3.2. We tried to test our model

using the dataset introduced by (Stanovsky et al., 2018) that is automatically generated from a Question Answering dataset, but we could not obtain it.

Experimental Results and Analysis

Three evaluation metrics were used to measure the performance of our model: Recall (R), Precision (P), and F-measure (F). All the aforementioned measures were expressed as percentages throughout the experiments. With the F-measure being the breakthrough performance measure. Detailed results of the experiments can be found in Table 3.3. In the next paragraphs, we apply a series of experiments to gain further insights on our proposed OIE model. To avoid penalizing different but equally valid arguments we follow the approaches described in Section 3.2 of (Cui et al., 2018) and (Stanovsky et al., 2018), by employing a partial matching strategy that permits variability in the predicted tuples. An argument or predicate is assumed valid if it partly matches the benchmark data above a certain threshold. This matching function was first applied by (He et al., 2015), where the authors consider the argument correct only if it contains the syntactic head of the gold argument, and the same applies to the predicate.

Table 3.3: Precision (P), Recall (R) and F-measure (F) scores of different neural models on Newswire and WikiNews datasets.

Network architecture	**Embeddings**	**Newswire**			**WikiNews**		
		P (%)	**R (%)**	**F (%)**	**P (%)**	**R (%)**	**F (%)**
BiLSTM	(GloVe)	41.1	45.1	43.0	47.3	46.9	47.1
BiLSTM	(ELMo)	43.0	46.7	44.8	49.6	50.7	50.1
GRU	(GloVe)	41.0	42.6	41.8	40.8	45.1	42.8
HAN	(ELMo)	32.2	43.3	36.9	37.5	45.9	41.3
BiGRU	(GloVe)	51.7	49.1	50.4	58.4	54.0	56.1
BiGRU	**(ELMo)**	**53.0**	**51.4**	**52.1**	**60.1**	**57.2**	**58.7**

Experiment 1

In the first experiment, we compare the results of employing ELMo embeddings against GloVe embeddings. As shown in Table 3.3, when a Bi-GRU network is employed using ELMo instead of GloVe, it yields an increase in the F-Measure from 56.1% to 58.7% on the WikiNews dataset and from 50.4% to 52.1% on the Newswire dataset.An increase in the F-Measure by 3% can also be observed when a Bi-LSTM model that uses contextual embeddings is

employed in contrast to non-contextual embedding. Hence, contextual embeddings notably affect the performance of the OIE task. The results demonstrate that the contextual representations essential captures more information, as it allows a more thorough interpretation of the linguistic characteristics of the dataset. An interpretable cause of the performance difference between ELMo and GloVe is that our extractions that form the relation triples are composed of more than one token, such as 'Barack Obama', and in such cases, contextual embeddings deal better with complex sentence structure. In addition to the massive differences in the embedding dimensionality between ELMo — 1024 — and GloVe —- 100—. This conclusion is envisioned because the more dimensionality of the word vectors, the more semantic information can be retained in the resulting representation. This explains the fall in performance attained from GloVe.

Experiment 2

Subsequently, in the second experiment, We compare our OIE model (BiGRU (ELMo)) against the model proposed by (Stanovsky et al., 2018) (BiLSTM (GloVe)). Table 3.3 shows the effect of utilizing contextual word embedding in a Bidirectional GRU network on extracting relation triples. The proposed model achieved an F-Measure of 52.1% compared to 43.0% achieved by (Stanovsky et al., 2018) on the Newswire dataset. Results on the WikiNews dataset followed the same trend, and our model increased the F-measure by 11.6%. It is observed that the proposed OIE system outperforms the model proposed by (Stanovsky et al., 2018) for the same reasons mentioned above in Experiment 1. Performance analysis of a model is a non-trivial task, as it is crucial to analyze when the model is unsuccessful. We inspect a random set of outputs from our model to identify major causes of errors. An important observation is the sentence length, as we notice that our proposed model is not capable of labeling the correct tuples when the sentence is relatively short, which results in the extraction of partial arguments and relations deemed incomplete or uninformative triples. This is probably because a more extended context offers richer information for disambiguation. Further, we find that another common error of our proposed model is sentences that contain unseen predicates during training.

Experiment 3

In the final experiment, we illustrate the effect of implementing a Bidirectional GRU instead of a single-direction GRU network. As we previously mentioned

in Section 3.3, unidirectional networks can only have access to past information. Thus the output is based on what the network has previously learned, unlike bidirectional networks that can capture both past and future information. This elaborates the decrease in the F-Measure of the GRU network on the both datasets, respectively, compared to our proposed Bidirectional GRU model.

It is noteworthy that we tested the effect of employing a Hierarchical Attention Network (HAN) (Yang et al., 2016b). Designed initially to classify reviews, HANs are a document classification model based on RNNs, where they are composed of several hierarchies in a way that the lower hierarchies feed the upper ones. Each hierarchy of the HAN consists of a dynamic BiLSTM or BiGRU with an attention mechanism. To elaborate further, HANs employ stacked RNNs on the word-level to capture the informative words in a sentence. It then combines the representation of those vital words to produce a sentence vector (Yang et al., 2016b). Nonetheless, our proposed OIE model under-performed using HANs. Our interpretation behind this is that HANs are not the appropriate strategy for our OIE task due to the structure of the datasets used, as the sentences that constitute both datasets — Newswire and WikiNews — are not correlated. This is grounded on the intuition of how HANs operate, as they rely on the fact that documents are made up of sequences of meaningful sentences, and these sentences consist of sequences of meaningful words. Our interpretation aligns with the findings of (Gao et al., 2018). In addition, the authors also claim that the performance of HANs is highlighted when the dataset size is large enough to allow the HAN to learn the linguistic patterns better, this is also another limitation of the Newswire and WikiNews datasets.

3.5 Conclusion

The Bidirectional GRU-based OIE model with contextual word embeddings presented in this study delivers higher performance than the existing state-of-the-art algorithm. The impact that contextual embedding had on our OIE architecture is notable in our experiments. In addition, Bidirectional GRU enhanced the performance with less complexity when compared to Bidirectional LSTM.

We believe there is still room for development in the field of OIE. OIE cannot be regarded as a solved NLP task. For instance, little work has been done in extracting N-ary relations. The main focus has been directed toward the extraction of binary relations, omitting the importance of higher-order

relations. The presented work can be further extended to extract the N-ary relation. In the future, we would like to test our model on a larger dataset and the model's adaptability to other languages. Finally, this approach can be employed in other NLP tasks such as question answering and summarization.

4 Transfer Learning for Open Information Extraction

Various tasks in natural language processing (NLP) suffer from a lack of labeled training data, which deep neural networks are hungry for. In this study, we relied upon features learned to generate relation triples from the open information extraction (OIE) task. First, we studied how transferable these features are from one OIE domain to another, such as from a news domain to a bio-medical domain. Second, we analyzed their transferability to a semantically related NLP task: relation extraction (RE). We thereby contribute to answering the question: can OIE help us achieve adequate NLP performance without labeled data? Our results showed comparable performance when using inductive transfer learning in both experiments by relying on a minimal amount of the target data, wherein promising results were achieved. When transferring to the OIE bio-medical domain, we achieved an F-measure of 78.0%, only 1% lower when compared to traditional learning. Additionally, transferring to RE using an inductive approach scored an F-measure of 67.2%, which was 3.8% lower than training and testing on the same task. As a result, our analysis shows that OIE can act as a reliable source task.

This work was originally published as:
Sarhan, I. and Spruit, M. (2020). Can we survive without labelled data in NLP? Transfer learning for open information extraction. *Applied Sciences*, 10(17):5758

4.1 Introduction

In deep learning for natural language processing (NLP), the collection of labeled data necessary for training and building models is expensive. This has further highlighted the urgency towards transfer learning research. Transfer learning aims to benefit from information gathered from previous training data in directly making predictions in the target task by utilizing the extracted information. Deep learning approaches in NLP did not start until the early 2000s (Otter et al., 2021). Recently, there has been an exponential increase in the number of scientific publications in neural networks in various NLP tasks (Otter et al., 2021).

Open information extraction (OIE) is a challenging task of extracting relation tuples from an unstructured corpus. Its main objective is to generate structured information from unstructured data in the form of a relation triple, ≪Argument 1 - Relation - Argument 2 ≫, without the need to predefine the relation between the two arguments. The extracted tuples can be binary, ternary, or n-ary, where the relationship is expressed between more than two entities such as the Person–Location–BornIn–BornOn relation ≪ *Jack Adams, Michigan, California, 1975* ≫.

Table 4.1: An example of extracting structured data from a given sentence to demonstrate how Open Information Extraction (OIE) differs from Relation Extraction (RE).

Sentence	John Lennon was born on 9 October 1940, in Liverpool and gained worldwide fame as the founder of the Beatles.
OIE Tuples	≪ John Lennon, Born, 9 October 1940 ≫ ≪ John Lennon, Born, Liverpool ≫ ≪ John Lennon, founder, Beatles ≫
RE Tuples	Person-Born-On: ≪ John Lennon, Born, 9 October 1940 ≫ Person-Born-In: ≪ John Lennon, Born, Liverpool ≫ Person-Organization: ≪ John Lennon, founder, Beatles ≫

Relation extraction (RE) — also classified as a category of information extraction — is the process of identifying semantic relationships between entities. Contrary to OIE, RE requires predefining the relationship prior to extraction. Similar to OIE, the extracted relation can either be a binary relation, for instance, Located-In ≪ *Berlin, Germany* ≫, or a higher order relation — n-ary —, for instance, a 3-ary relation between Employee–Position–Company ≪ *Adam Smith, Marketing Manager, XYZ Company* ≫ Examples of both OIE and RE triples can be found in Table 4.1.

OIE is a crucial NLP task, and thus it was chosen as a source task to transfer to other NLP tasks due to its various potential applications in information retrieval, information extraction, text summarization, and question answering (Mausam, 2016). While various OIE algorithms have been developed in the past decade, only a small number employ deep learning techniques.

In recent years, researchers have increasingly been showing interest towards model generalization in deep learning due to the lack of labeled data. In this study, we investigated the ability to transfer OIE to other NLP tasks, ranging from domain–adaptation (news domain to bio-medical) to RE as a semantically related task. RE task was chosen because of the nature of both OIE and RE, and the semantic overlap between both tasks backed up our choice. We also compared and experimented with different word embeddings throughout our research.

This work aimed to measure how OIE can assist in other NLP tasks. Our primary objective was to conduct a fair comparison of different methods and settings with respect to OIE transfer learning effects on other NLP tasks. Therefore, we did not focus on outperforming state-of-the-art results in the target tasks.

The remainder of this chapter is structured as follows. Section 4.2 presents a brief overview of transfer learning, while Section 4.3surveys previous work in both OIE and RE. The neural network architecture is explained in Section 4.4, and the experimental setup is explained in Section 4.5. Results and evaluation are discussed in Section 4.6. Finally, Section 4.7 concludes the chapter and discusses potential future work for this study.

4.2 Transfer Learning in NLP

Formerly, there was a misconception that a machine learning framework would achieve the desired results only if the testing and training data had similar distribution and feature space. Thus, a new framework was required for data with different distribution properties and features, making the collection of labeled training data expensive and complicated. Transfer learning lessens the demand of gathering an immense amount of labeled training data by reemploying the knowledge gained from a different task to tackle new tasks faster and constructively.

Pan and Yang introduced a transfer learning taxonomy (Pan, 2020). Additionally, they categorized transfer learning into three classes:

- *Inductive transfer learning:* labeled data are accessible in the source and

target domain.

- *Transductive transfer learning:* labeled data are only available in the source domain.

- *Unsupervised transfer learning:* No labeled data available in either the source or target domain.

Transfer learning has been implemented in various different machine learning tasks, achieving notable results, for instance, textual summarization (Keneshloo et al., 2019), named entity recognition (Bhatia et al., 2020), question answering (Min et al., 2017; Yu et al., 2018), and text classification (Do and Ng, 2005).

BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2019b) was a breakthrough in transfer learning on a range of language-based tasks, not only due to the fact that BERT was pre-trained on an immense dataset but also because it has a substantial number of transformer blocks (encoder layers) and feed-forward networks. Later on, many transfer learning models built on BERT were introduced, for example, ULMFiT (Howard and Ruder, 2018) and OpenAI transformer (Radford et al., 2018) . This novel development also affected how words are encoded, with more elaboration being found in Section 4.4.

As shown in Figure 4.1, two transductive transfer learning experiments were carried out in our work. The first one transfers knowledge learned from the OIE news domain to the OIE bio-medical domain, referred to as domain adaptation. In contrast to transfer learning, domain adaption entails adapting a model trained on one domain to other different domains on the same task. The default process of supervised domain adaptation for neural models involves pre-training the network on data from the source domain, followed by fine-tuning hyperparameters on data from the target domain. The second experiment transfers information from the OIE news domain to the RE news domain. Moreover, a small percentage of OIE bio-medical data was added to OIE news data to experiment with inductive transfer learning. Similarly, a small amount of RE training data were inputted into the neural model along with the OIE news corpus, with both experiments being referred to as multi-task learning.

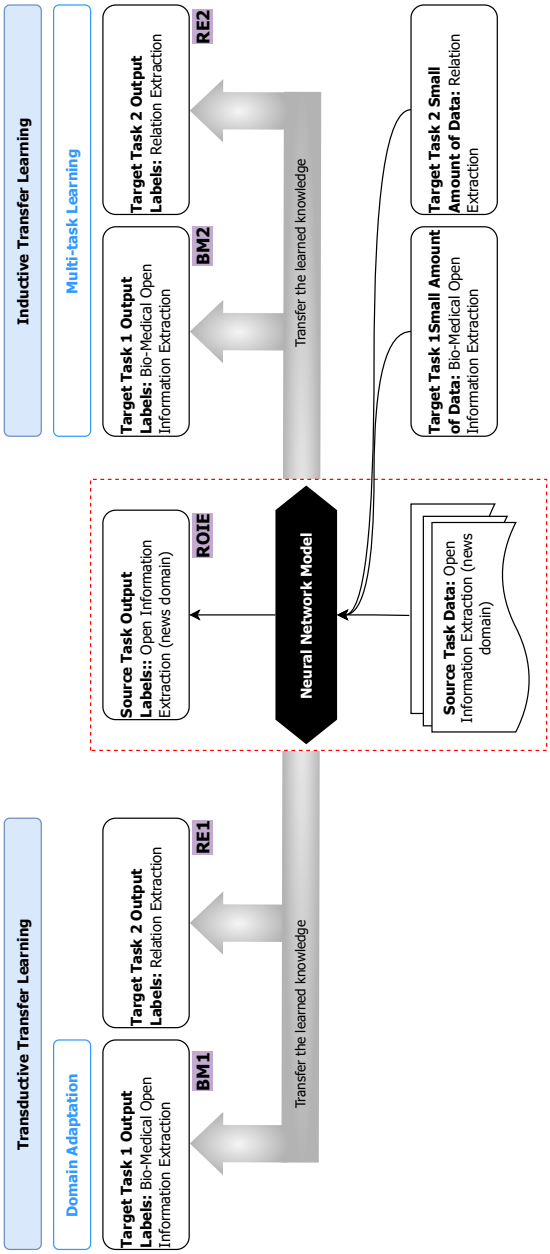

Figure 4.1: Open information extraction (OIE) transfer learning assessment. A total of four transfer learning experiments were carried out in our work. Left: the two transductive transfer learning experiments (BM1 and RE1). Right: illustration of the two experiments using inductive approaches (BM2 and RE2). Middle: the red dotted line represents the original model (ROIE), in which we tested our proposed neural model by testing and training on OIE news data, which is discussed in Section 4.4. The experiments' ID and the section they are discussed in are encapsulated in the yellow rectangles.

4.3 Related Work

This section focuses on previous works performed on OIE and RE relation extraction in the literature.

State-of-the-Art on Open Information Extraction

OIE can be portrayed in three broad categories as elaborated in Chapter 2: (a) machine learning classifier approaches, (b) hand-crafted rules approaches, and (c) neural network approaches. The first two categories can be further divided into two sub-categories: shallow syntactic analysis and dependency parsing. Below we discuss state-of-the-art work in each of these categories.

Machine Learning Classifiers Approaches

OIE systems that are built on machine learning classifier techniques require automatically generated data to train the classifier. In 2008, Banko et al. introduced the first OIE system based on shallow syntactic analysis, TextRunner (Etzioni et al., 2008). It implements extraction in three main phases. It starts with a self-supervised learner that depends mainly on a conditional random field (CRF) classifier that utilizes unlexicalized features required for relation extraction, followed by a single pass extractor that extracts any potential relation triple and classifies each as either trustworthy or not. Finally, a redundancy-based assessor re-ranks the extracted relations and assigns a confidence score to each extracted tuple is implemented. Not only did the authors of TextRunner facilitate domain-independent detection of relations from a corpus, but their work triggered researchers toward developing OIE systems. For instance, the WOE (Wikipedia-based Open Extractor) (Wu and Weld, 2010) system is built on TextRunner, having two modes of operation: WOE^{POS} and WOE^{Parse}. The central hypothesis behind WOE is the automated assembly of training samples by heuristically pairing Wikipedia info box values with corresponding texts, hence improving TextRunner's performance. WOE^{POS} exploits the CRF classifier trained with shallow syntactic proprieties to extract specific words between two noun phrases representing a relation.

An example of an OIE approach that utilizes dependency parsing is WOE^{Parse}; it exploits a rich dictionary of dependency path patterns acquired from Wikipedia extractions. While the OLLIE (Open Language Learning for Information Extraction) approach (Schmitz et al., 2012) relies on the bootstrapping concept, it learns semi-lexicalized pattern templates using dependency parses by bootstrapping a plentiful amount of training data that results in surpassing WOE's

performance.

Hand-crafted Rules Approaches

ReVerb, introduced by (Fader et al., 2011), extracts tuples by singling out relation phrases that satisfy syntactic and lexical constraints; for each relation phrase, a pair of noun phrase arguments are identified. ReVerb then uses logistic regression trained on 1000 sentences from the web with shallow syntactic features to assign a confidence score to each extracted relation triple. The R2A2 approach (Christensen et al., 2011b) upgrades ReVerb by adding ArgLearner, an argument identifier that makes use of patterns as features to identify the left and right boundaries of each argument.

KRAKEN (Akbik and Löser, 2012) is one of the few OIE systems that are able to capture N-ary relations. It utilizes hand-crafted patterns to identify relation phrases and their correlated arguments over typed dependency parsers. As a further matter, KRAKEN is able to detect the completeness and correctness of the extracted facts, thus increasing the quality of the extracted information. Del Corro and Gemulla proposed ClausIE (Clause-based Open Information Extraction) (Del Corro and Gemulla, 2013), which locates clauses in input sentences by making use of linguistic information of the English language's grammar by computing a dependency parse tree of the input phrase to determine its syntactical structure. Each clause is later classified to be compatible with the grammatical function of its constituents. Unlike the aforementioned OIE systems, ClausIE does not exploit any training data.

Neural Network Approaches

Recently, as a result of their success in diverse NLP tasks (Otter et al., 2021), deep neural networks paved the way for the OIE task. A recurrent neural network (RNN) encoder-decoder OIE framework was proposed by (Cui et al., 2018). A fluctuating length sequence is sent to the network's encoder as a sole input. The encoder then generates a compressed representation vector to transfer to the decoder in order to produce the output sequence. A three-layer long short-term memory (LSTM) (Hochreiter and Schmidhuber, 1997) is the internal structure of both the encoder and the decoder. (Stanovsky et al., 2018) presented a neural OIE paradigm that trains a bidirectional LSTM (bi-LSTM) transducer to label each word, verifying that supervised learning can have a positive effect on OIE performance.

State-of-the-Art on Relation Extraction

RE research falls mainly under one of the following approaches: supervised, semi-supervised, distant supervision, and unsupervised. As always, the main issue of supervised techniques is the necessity of having a large amount of labeled data, which is challenging to gather (Sarhan et al., 2016b). Semi-supervised approaches mainly depend on bootstrapping techniques. Distant supervision techniques merge both semi-supervised and unsupervised approaches. However, the popularity of unsupervised techniques declined due to the fact that the learner is provided unannotated data, and for that reason, evaluation becomes demanding at a large scale. We limited our discussion to supervised, semi-supervised, and distant supervision approaches. Neural approaches appear as a subclass in all the aforementioned classes.

Supervised Approaches

RE is treated as a multi-class classification task in supervised approaches. Supervised categories can be classified into kernel-based approaches and feature-based approaches. An example of the latter is the work of (Zhou et al., 2005), who merged diverse lexical, syntactic, and semantic knowledge features by employing a support vector machine (SVM) to extract relations, proving the effectiveness of base phrase chunking information. Authors of (Plank and Moschitti, 2013) introduced a kernel-based RE paradigm that incorporates term generalization techniques—word clustering and latent semantic analysis—with structured kernels to enhance RE results in different domains. Moreover, Su and Vijay-Shanker (2020) proposed a neural approach based on adversarial training, aiming to boost RE task performance through various adversarial examples and adding perturbation on all input features of the model. Adversarial learning is built on the basis that similar data instances are assigned the same label.

Semi-Supervised Approaches

The first bootstrapping algorithm was DIPRE (Dual Iterative Pattern Relation Expansion) (Brin, 1999), which employs a pattern-matching model as a classifier by using a set of seeds to extract patterns from the dataset in order to extract new candidate relations. The DualRE model (Lin et al., 2019) was proposed to overcome the problem of semantic drift associated with bootstrapping approaches. The key idea behind DualRE is training a retrieval module along a relation prediction module, as a result of this, mutually improving the quality of one another through labeling data to use as auxiliary training

data. In (Welling and Kipf, 2016), a convolutional neural network (CNN) RE architecture was proposed that employs graph-structured data where label knowledge is smoothed over the graph by means of explicit graph-based regularization.

Distant Supervision Approaches

The traditional distant supervision RE approaches claim that if a sentence consists of two related entities, then the same relation lies between those two entities. For instance, in their work on relation classification, Mintz et al. (2009) utilized distant supervision and included various textual features such as lexical and syntactic features, named entity tags, and conjunctive features. Their approach relied heavily on feature engineering, as is common in traditional methods. On the other hand, Sebastian et al. proposed an RE model that supports a different claim, "if two entities participate in a relation, then at least one sentence that mentions those two entities might express that relation" (Riedel et al., 2010), by utilizing a factor graph to aid in determining if two entities are related or not. Additionally, a learning algorithm is employed to train this graphical framework by structuring distant supervision as an instance of constraint-driven semi-supervision.

A piecewise CNN RE technique was proposed by (Zeng et al., 2015) not only to overcome the noise generated from the feature extraction phase but also to address the issue of handling distant relation extraction as a multi-instance task, which leads to a lack of certainty of instance labels. By designing a convolutional framework with piecewise max pooling as an alternative to feature engineering to learn related features automatically, the authors of (Zeng et al., 2015) were able to overcome the aforementioned problems.

4.4 ROIE: A Recurrent Neural Network Model for Open Information Extraction

Our recurrent neural network (RNN) model is based on our work in Chapter 3 by tackling the OIE task as a sequencing labeling problem resulting in the extraction of multiple, overlapping tuples for each sentence.

Neural Model Architecture

RNNs are prone to vanishing and exploding gradient descent complications throughout the back-propagation process, making RNN training challenging.

Thus, LSTMs and gated recurrent units (GRUs) were established to address the issues related to the unstable gradient. When the gradient becomes too big or simply disappears, killing the learning process, LSTMs and GRUs aid by using the relevant gates to allow the gradient to flow backward through time, freely and effectively keeping long-term dependencies (Pascanu et al., 2013).

LSTMs and GRUs can train on extended word contexts and connect information using cell states. LSTM has three gates (input, output, and forget), contrary to GRU, which couples the input and forget gates in one gate—the update gate, in addition, to the reset gate, which determines how to incorporate previous memory with the current input. As a result, our model employs GRUs instead of LSTMs, since GRUs are less complex with only two gates. With this, they require fewer training parameters and utilize less memory, effectively making GRUs faster than LSTMs.

The default operation in RNN captures the context in a single direction, which may lead to comprehending issues; for instance, consider the following two sentences:

- Second place is not as prestigious as first place.
- Second is the standard international unit of time.

In these sentences, the word "second" carries different meanings, which traditional RNNs will not be able to comprehend since it is the first word in the sentence; nevertheless, bidirectional RNNs support learning from both ends. A bidirectional GRU (Bi-GRU) was employed in our model to learn the forward and backward lexical semantics of each word in a given sentence. There are two different methods to implement a bidirectional network, either by having two RNNs operating in opposite directions or within the internal architecture of the RNN itself. In our ROIE framework, we implemented the latter approach.

Word Embeddings

Recently, several types of word embeddings have been introduced; nevertheless, they all serve the same purpose of mapping words to low-dimensional vector representations. The aforementioned OIE and RE deep learning-based approaches in Section 4.3 and Section 4.3, respectively, utilized one of the traditional word embeddings, either GloVe (Pennington et al., 2014) or Word2Vec (Mikolov et al., 2013b).

In our work, we incorporated novel contextualized word embeddings. Due to their ability to capture complex syntactic and semantic features of a word, deeply contextualized word embeddings have proven successful in various NLP

Table 4.2: Word embeddings employed in our work.

Embedding	Dimensionality	Trained On
GloVe	100	Aggregated global word-word co-occurrence statistics from a corpus.
BERT	3072	Wikipedia and +10000 books of different genres.
XLNet	2048	Over 130 GB of textual data.
XLM-RoBERTa	1024	2.5 TB of filtered CommonCrawl data.

tasks compared to traditional word embeddings. The main concept behind contextualized word embeddings is that a word's representation varies according to its neighboring words. Thus the same word can have different representations depending on its adjacent words.

Table 4.2 shows the word embeddings we employed in our experiment, along with the dimensionality of each embedding and the data they are trained on. We picked one traditional non-contextualized embedding, GloVe, and three contextualized embeddings with different dimensionalities: BERT (Devlin et al., 2019b), XLNet (Yang et al., 2019), and XLM-RoBERTa (Conneau et al., 2020b). XLNet is trained on data much larger than Google's BERT training data, and thus it outperforms BERT on 20 different NLP tasks (Yang et al., 2019). Facebook's XLM-RoBERTa depends on the masked language model objective and is effective in text processing from 100 different languages.

Flair (Akbik et al., 2019) is a simple framework that offers a unified interface for conceptually varying types of word and document embeddings, which we utilized in our experiments.

Work Flow

The embedded sentence—composed of a fixed-length vector—is sent as an input to our ROIE neural network framework. Specifically, predicates—the part of a sentence or clause containing a verb and stating something about the subject—are regarded as the building blocks of most languages, as they denote significant actions that are deemed extremely efficient in extracting relations of interest. Therefore, in line with the work of (Stanovsky et al., 2018) and the work shown in Chapter 3, the predicate in each sentence is presumed to be the relation that links the tuple; consequently, the predicate is inputted to the neural network framework as a feature vector alongside the part of speech (POS) tag of the input sentence obtained using the NLTK toolkit (Bird, 2006), as shown in Figure 4.2.

After embedding the three aforementioned inputs, we concatenated them all to form our feature vector of shape (3, length of sentence, embedding size);

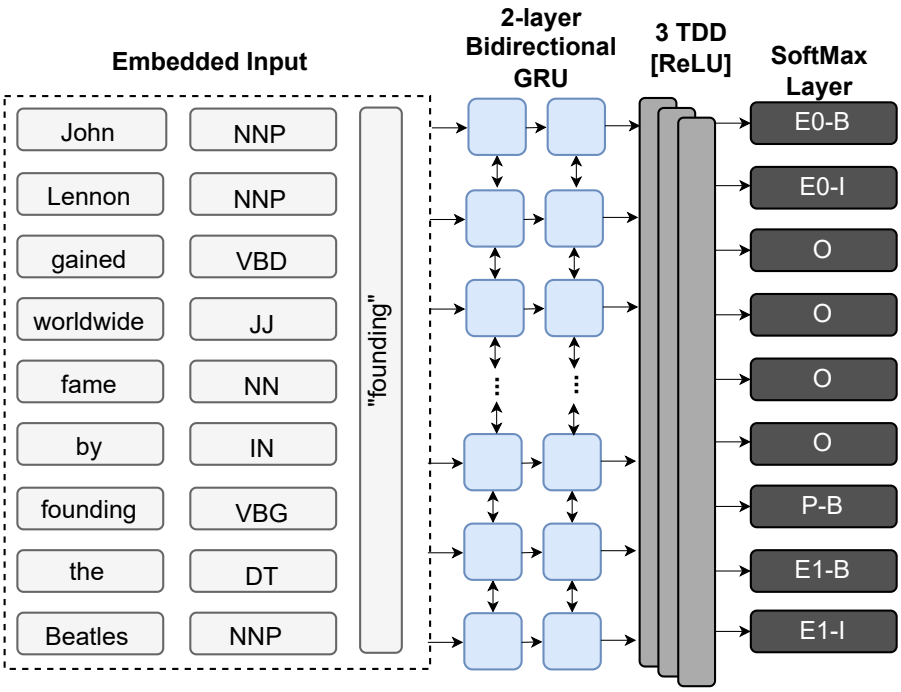

Figure 4.2: Our ROIE transferable neural model architecture.

the feature vector is defined as follows:

$$FeatureVector = Emb(Word) \bigoplus Emb(POS) \bigoplus Emb(Predicate) \quad (4.1)$$

The generated feature vector is then passed to the two-layer Bi-GRU, which in turn outputs a tensor that is progressed to three-layer time-distributed dense (TDD) layers, which is finally passed to the SoftMax layer for label prediction.

Sequence Labelling

In NLP, sequence labeling is the task of identifying and assigning a label to each word, for instance, the POS task, where each word is tagged to a particular POS. Sequencing labeling achieved more promising results when compared to traditional statistical techniques among a diverse array of NLP tasks (Stanovsky et al., 2018). In our work, we used BIO tags (Begin-Intermediate-Outside) (Ramshaw and Marcus, 1995) to indicate the word's location in the sentence and label it accordingly. The SoftMax output layer assigns the probability score to each word to determine its corresponding label, as shown in Figure 4.2. Our proposed ROIE paradigm is only able to capture binary relations. If a sentence contains more than one predicate, another instance of that sentence is created to capture any possible relation. However, if a sentence has no relations, only the predicate is labeled "P-B" (Predicate-Begin) and "P-I" (Predicate-Intermediate), while the label "O" (Outside) is assigned to the remaining words in the sentence without assigning any "E" (Entity) labels.

Dataset

To train and test our OIE neural framework, we used the Wikipedia News Corpus (WikiNews) (Stanovsky and Dagan, 2016). Our dataset was split into a training set to train the network, a development set for validation purposes, and a test set to assess the performance of our ROIE framework on a 60/20/20 ratio. An overview of the dataset is shown in Table 4.3.

Hyperparameter Settings

Our ROIE neural framework was implemented using the Keras framework (Chollet, 2015) with a TensorFlow backend (Abadi et al., 2016). Table 4.4 shows our model's hyperparameter configurations that achieved the best results when training and testing on OIE. As shown, our framework was trained

Table 4.3: WikiNews dataset overview.

Dataset	No. of Sentences	No. of Tuples
Train Set	1174	2906
Development Set	392	946
Test Set	393	993

Table 4.4: Hyperparameter settings used in ROIE.

Hyperparameter	Value
Epochs	20
Batches	100
Bidirectional GRU	128 units
TDD Activation Function	ReLU
TDD units	128 units
Dropout Rate	0.1
Optimizer	Adam

on 20 epochs, and the training dataset was split into 100 batches. For regularization purposes, in order to avoid over-fitting, the dropout rate was set to 0.1. Furthermore, early stopping was utilized to terminate training when the training performance stopped improving. Both bidirectional GRU layers and the three TDD layers had an equal number of 128 units. Additionally, rectified linear unit (ReLU) (Nair and Hinton, 2010a) was the chosen activation function in the three TDD layers, while the Adam optimizer (Kingma and Ba, 2015) was utilized for training our framework.

Results of our ROIE Model

It should be emphasized that our ROIE neural model outperformed other state-of-the-art neural OIE approaches, as documented in Chapter 3, while using ELMo word embeddings (Peters et al., 2018), also a deep contextualized word embedding that models both complex syntactic and semantic features of a word.

Better results were attained after XLNet was substituted for ELMo (Peters et al., 2018) when compared to our results in Chapter 3; the results are reported in Table 4.5. An exhaustive grid search was performed to single out the best batch–epoch pair for each word embedding. Our batches and epochs ranged from 20 to 120 and 1 to 50, respectively, both with increments of 5. GloVe

achieved an F-measure of 56.1%, while BERT and XLM-RoBERTa achieved an F-measure of 61.1% and 61.5%, respectively. Nevertheless, XLNet surpassed all the other embeddings—including ELMo's 59% F-measure—and achieved 65%.

4.5 Material and Methods

In this section, we explain the experiments carried out and the dataset utilized in our two main tasks, transferring to the OIE bio-medical domain and the RE task. In the source task, the aforementioned WikiNews training set (Stanovsky and Dagan, 2016) was utilized.

Transferring to OIE: Bio-Medical Domain

A classifier trained on a news corpus would observe an altered distribution if employed to classify bio-medical data. Therefore, domain adaptation methods are deployed in transfer learning in such scenarios. In the transductive learning task, specifically domain adaptation, we handle our pre-trained model as a feature extractor; in our case, the pre-trained model was trained on the news domain, where there is a characteristic shift in the distribution of the data between source and target domains that necessitates adjustments to transfer knowledge effectively.

DDIExtraction 2013 (Segura-Bedmar et al., 2011) is a bio-medical dataset mainly specialized in the subject of drug-drug interactions. The dataset was structured from the DrugBank database and MEDline abstracts related to drug-drug interactions (Wishart et al., 2006). We utilized the DDIExtraction as a test set in the following experiments. In our work, the performance of the following three experiments was compared against each other:

1. *Transductive transfer learning:* transferring knowledge learned from the OIE news domain to the OIE Bio-medical domain.

2. *Inductive transfer learning:* A small amount of Bio-medical data also from DDIExtraction is fed to the neural network alongside news data to train the neural network.

3. *Traditional learning:* Both training and testing on Bio-medical data.

Table 4.5: Results of the ROIE model using different word embeddings. Both training and testing are done on the OIE WikiNews dataset. Recall (R), precision (P), and F-measure (F) are used as evaluation metrics.

Source Task (Train)	**Target Task (Test)**	**Word Embeddings**	**Hyperparameters (Batches -Epochs)**		**Results (R-P-F)**		
OIE (news)	OIE (news)	GloVe	100	5	58.2%	54.1%	56.1%
		BERT	100	5	64.3%	58.2%	61.1%
		XLNet	**100**	**20**	**68.1%**	**62.2%**	**65.0%**
		XLM-RoBERTa	100	5	65.4%	58.1%	61.5%

Table 4.6: List of predefined relations in the SemEval-2010 corpus and their number of occurrences.

Relations	Number of instances	
	Train set	**Test set**
1. Cause-Effect	485	228
2. Instrument-Agency	245	156
3. Product-Producer	320	231
4. Entity-Origin	398	258
5. Entity-Destination	392	252
6. Component-Whole	209	110
7. Content-Container	118	102
8. Member-Collection	345	233
9. Message-Topic	394	261
Total	**2906**	**1831**

Transferring to Relation Extraction

The OIE and RE tasks are both subclasses of information extraction, making the two tasks similar in semantics. The dataset used in the RE task for training, testing, and validation is Semeval-2010 Task 8 (Hendrickx et al., 2009). The nine predefined relations in the dataset are shown in Table 4.6. The training set consists of 8000 sentences. However, for a fair comparison, we trained our neural network on the same number of relation tuples available in the OIE training set; thus, 2906 tuples were randomly selected from the training set. Similarly, the same experiments were compared against each other when transferring from the OIE news domain to RE:

1. *Transductive transfer learning:* transferring knowledge learned from the OIE news domain to the RE news domain.

2. *Inductive transfer learning:* A small percentage of the RE corpus is fed into the neural framework along with OIE news data to train the neural network.

3. *Traditional learning:* Both training and testing on the RE news domain.

In all the above-mentioned experiments in both tasks, we used bio-medical OIE and RE, a development set containing 946 tuples composed of the same structure as the source task, for validation purposes.

4.6 Results and Evaluation

The following measures were used to measure the effect of transferring knowledge learned from our ROIE framework: Recall (R), Precision (P), and F-measure (F). All the aforementioned evaluation metrics were expressed as percentages throughout the experiments, with the F-measure being the determining performance measure. All hyperparameters —shown previously in Table 4.4 — except for epochs and batches were fixed throughout our experiments. Contextual embeddings were highly sensitive to changes in hyperparameters, specifically with respect to the number of epochs and batches. Steep falls and rises were noticed when the number of epochs and batches were changed.

It is worth noting that the dimensionality of the word embeddings refers to the length of the vector; in theory, the size of the vector is directly proportional to the information it can store, which allows NLP systems to perform better. However, in practice, there was not much benefit with the embeddings with higher dimensionality when compared with lower dimensionality embeddings.

Results of Transferring to OIE: Bio-Medical Domain

In order to properly evaluate transfer learning results, we compared them with training and testing on the target task. Detailed results of the experiments can be found in Table 4.7, indicating the source task (training set) and the target task (testing set). The hyperparameters that achieved the highest scores are the ones reported in Table 4.7.

OIE: Bio-Medical Domain Results Discussion

Our system achieved the highest results using XLM-RoBERTa in all three experiments: transductive transfer learning, inductive transfer learning, and traditional learning, outperforming all other word embeddings.

When our training set was composed entirely of news data, XLM-RoBERTa scored the highest F-measure of 64.4% with 100 batches and 5 epochs. XLNet and GloVe achieved the same F-measure of 62.9% using the same number of batches and epochs, 100 and 5, respectively. Nevertheless, BERT achieved the lowest F-measure of 60%.

In inductive transfer learning, a small amount of bio-medical data were inputted to the neural framework by sampling a random batch from the DDIExtraction 2013 training data using a 4:1 ratio, with bio-medical data having the lower ratio. A significant increase in the F-measure of 13.6% was attained in

inductive transfer learning compared to transductive transfer learning. Using both XLM-RoBERTa and XLNet, our inductive transfer approach realized an F-measure of approximately 78%, with XLM-RoBERTa's precision surpassing XLNet's by 0.9%. BERT came in third and achieved 75.2%, while GloVe scored an F-measure of 73.7%. This shows that our proposed ROIE model has the ability generalize across domains. Further, we find that the cross-domain capabilities of XLM-RoBERTa are to some extent better in the cross-domain evaluation compared to other word embeddings.

The results scored using traditional learning by training entirely on biomedical data were only 1% higher than those achieved using the inductive transfer learning technique. Once again, XLM-RoBERTa outperformed the other embeddings by scoring an F-measure of 79% using 100 batches and 15 epochs. Additionally, BERT achieved roughly the same F-measure as XLM-RoBERTa of 78.9%, using the same number of epochs and batches; however, it achieved a lower precision of 85.9%. It is notable that GloVe achieved a higher F-measure in inductive transfer learning than traditional learning. We interpret that adding news training data to the biomedical tasks resulted in higher performance with GloVe embeddings. This could correlate with the original training data of the GloVe model used in our experiments. Thus, our results show that using a small percentage from the target task while training our neural network results in a proximate outcome when compared to traditional learning.

Results of Transferring to Relation Extraction

Equally, in order to establish a fair comparison in the following three experiments, we fixed the training set size to 2906 relation instances. Results of both transductive and inductive transfer learning were compared against the results achieved by traditional learning. Results are reported in Table 4.8.

Relation Extraction Results Discussion

Firstly, in transductive transfer learning, with 50 batches and 10 epochs, BERT was able to achieve an F-measure of 54.4%. Both XLNet and XLM-RoBERTa scored the same F-measure of 49.1%, which was nearly 4.6% higher than the F-measure achieved using GloVe.

With inductive transfer learning, we found an improvement of 12.8% when compared to transductive learning, also using a 4:1 ratio, with the OIE news dataset overtaking the higher ratio. Using XLM-RoBERTa, a 67.2% F-measure was attained when the network was trained on 15 epochs, and the training

Table 4.7: Domain adaptation results by transferring from the OIE news domain to the OIE bio-medical domain using four different word embeddings. Bold values indicate the highest achieved F-measure in the three experiments (transductive transfer learning, inductive transfer learning, and traditional learning).

	Source Task (Train)	**Target Task (Test)**	**Word Embeddings**	**Hyperparameters (Batches - Epochs)**		**Results (R-P-F)**		
Transductive Transfer Learning (BM1)	OIE (news)	OIE (bio-medical)	GloVe	100	5	68.2%	58.4%	62.9%
			BERT	50	10	**72.4%**	51.3%	60.0%
			XLNet	100	5	68.4%	58.3%	62.9%
			XLM-RoBERTa	**100**	**5**	71.0%	**59.0%**	**64.4%**
Inductive Transfer Learning (BM2)	OIE (news) + OIE (bio-medical)	OIE (bio-medical)	GloVe	100	15	69.8%	78.2%	73.7%
			BERT	100	5	71.9%	78.9%	75.2%
			XLNet	100	10	**73.6%**	82.9%	77.9%
			XLM-RoBERTa	**100**	**5**	73.0%	**83.8%**	**78.0%**
Traditional Learning	OIE (bio-medical)	OIE (bio-medical)	GloVe	100	5	70.8%	71.7%	71.2%
			BERT	100	15	73.1%	85.9%	78.9%
			XLNet	100	15	72.9%	84.2%	78.1%
			XLM-RoBERTa	**100**	15	**75.2%**	**86.9%**	**79.0%**

dataset was divided into 100 batches. BERT and XLNet did not fall far behind XLM-RoBERTa, as they achieved F-measures of 66.3% and 65.4%, respectively. GloVe achieved the lowest F-measure of 59.9%.

When employing default learning settings, where we train on our target task, there was a 3.8% enhancement in the F-measure. Once again, BERT outperformed by scoring an F-measure of 71%, only 0.5% higher than XLNet, and 2.6% higher than XLM-RoBERTa. Consistently, GloVe scored the lowest F-measure of 65.9%, hereby proving the notable effect in the model's performance when using contextualized word embeddings in contrast with traditional word embeddings.

Table 4.9 summarizes the best results of the three main experiments acquired in our work: ROIE model, transferring to the bio-medical domain, and transferring to RE. As seen in Table 4.9, we could not single out a particular contextualized word embedding to utilize, as the use of word embedding may vary according to the various reasons: type of task (OIE, RE, or sentiment analysis), dataset domain (news, bio-medical data, or financial data), and the computational power available to the user. This is also in agreement with other papers that extensively compared embeddings in various tasks and found that the most suitable one is highly dependent on the task and data nature (Tawfik and Spruit, 2020; Perone et al., 2018)

To further elaborate that the choice of the word embedding is dependent upon the task and nature of data, XLNet outperformed all the other word embeddings when training and testing on the news dataset. However, on bio-medical data, XLM-RoBERTa performed better in all three experiments: transductive transfer learning, inductive transfer learning, and traditional learning. It is worth noting that XLM-RoBERTa outperformed four out of a total of seven experiments in our work. Thus, we were motivated to compare and experiment with the use of different word embeddings.

4.7 Conclusion

Can we survive without labelled data in NLP? On the basis of our findings: yes! Nevertheless, employing labelled data in NLP tasks still results in better performance. However, the process of collection of labelled data is demanding and, in some cases, inaccessible. In this chapter, we utilized training on OIE to diminish the complication of insuffcient training data of neural network models in various NLP tasks and encourage model generalization. Since OIE plays a fundamental role in turning massive, unstructured data into factual information that can be used as a foundation to many NLP tasks, we favored

Table 4.8: Results of transferring from OIE to RE using four different word embeddings. Bold values indicate the highest achieved F-measure in each of the three experiments (transductive transfer learning, inductive transfer learning, traditional learning).

	Source Task (Train)	**Target Task (Test)**	**Word Embeddings**	**Hyperparameters (Batches - Epochs)**		**Results (R-P-F)**		
Transductive Transfer Learning (RE)	OIE (news)	RE (news)	GloVe	100	5	55.9%	37.0%	44.5%
			BERT	**50**	**10**	**62.2%**	**48.4%**	**54.4%**
			XLNet	50	5	58.8%	42.1%	49.1%
			XLM-RoBERTa	100	15	53.2%	45.6%	49.1%
Inductive Transfer Learning (RE2)	OIE (news) + RE (news)	RE (news)	GloVe	100	10	52.8%	69.3%	59.9%
			BERT	100	5	**61.7%**	73.0%	66.3%
			XLNet	100	15	59.7%	72.2%	65.4%
			XLM-RoBERTa	**100**	**15**	59.7%	**76.9%**	**67.2%**
Traditional Learning	RE (news)	RE (news)	GloVe	100	15	57.6%	77.1%	65.9%
			BERT	**100**	**15**	**62.4%**	**82.3%**	**71.0%**
			XLNet	100	5	61.6%	81.3%	70.5%
			XLM-RoBERTa	100	15	59.8%	79.9%	68.4%

Table 4.9: Summary of the best result obtained in each experiment by different systems described in the chapter: original ROIE model, transferring from OIE to bio-medical OIE (transductive transfer learning, inductive transfer learning, traditional learning), and transferring from OIE to (transductive transfer learning, inductive transfer learning, traditional learning).

Source Task (Train)	**Target Task (Test)**	**Word Embeddings**	**Hyperparameters (Batches - Epochs)**		**Results (R-P-F)**		
OIE (news)	OIE (news)	XLNet	100	20	68.1%	62.2%	65.0%
OIE (news)	OIE (bio-medical)	XLM-RoBERTa	100	5	71.0%	59.0%	64.4%
OIE (news) + OIE(bio-medical)	OIE (bio-medical)	XLM-RoBERTa	100	5	73.0%	83.8%	78.0%
OIE (bio-medical)	OIE (bio-medical)	XLM-RoBERTa	100	15	72.5%	86.9%	79.0%
OIE (news)	RE (news)	BERT	50	10	62.2%	48.4%	54.4%
OIE (news) + RE (news)	RE (news)	XLM-RoBERTa	100	15	59.7%	76.9%	67.2%
RE (news)	RE (news)	BERT	100	15	62.4%	82.3%	71.0%

OIE as our source task, thereby ensuring our work is useful and beneficial to the NLP community.

In the domain adaptation experiment, we transferred information learnt from one domain to the other on the same task. The neural model was trained on the OIE news domain and tested on the bio-medical domain. Results obtained from the inductive approach indicated that our ROIE neural model can play a fundamental role in domain adaptation.

Moreover, our research also covered the transferability to a semantically related task. Results achieved from transferring from the OIE to RE followed the same pattern as transferring from the OIE news domain to the bio-medical domain. Inductive transfer earning achieved promising and comparable results with traditional learning. Thus, our work demonstrates that OIE can act as a reliable source task, not only in domain adaptation but also when transferring to related tasks.

In the future, we intend to expand our work beyond sequence labelling tasks and experiment with multi-transfer learning thoroughly on several NLP tasks, specifically tasks that are not semantically related to OIE, as we believe that it is worth exploring its potential for generalizability through further experimentation on other tasks such as sentiment analysis. Additionally, we intend to investigate different transferring mechanisms to study how to leverage knowledge acquired from pre-trained models in varied ways.

5 Enhancing the Generalizability of Language Models

This chapter presents our strategy to address the SemEval-2022 Task 3 PreTENS: Presupposed Taxonomies Evaluating Neural Network Semantics. The goal of the task is to identify if a sentence is deemed acceptable or not, depending on the taxonomic relationship that holds between a noun pair contained in the sentence. For sub-task 1 —binary classification— we propose an effective way to enhance the robustness and the generalizability of language models for better classification on this downstream task. We design a two-stage fine-tuning procedure on the ELECTRA language model using data augmentation techniques. Rigorous experiments are carried out using multi-task learning and data-enriched fine-tuning. Experimental results demonstrate that our proposed model, UU-Tax, is indeed able to generalize well for our downstream task. For sub-task 2 —regression— we propose a simple classifier that trains on features obtained from Universal Sentence Encoder (USE). In addition to describing the submitted systems, we discuss other experiments that employ pre-trained language models and data augmentation techniques. For both sub-tasks, we perform error analysis to further understand the behaviour of the proposed models. We achieved a global $F1_{\text{Binary}}$ score of 91.25% in sub-task 1 and a rho score of 0.221 in sub-task 2.

This work was originally published as:
Sarhan, I., Mosteiro, P., and Spruit, M. (2022). UU-tax at SemEval-2022 task 3: Improving the generalizability of language models for taxonomy classification through data augmentation. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 271–281, Seattle, United States. Association for Computational Linguistics

5.1 Introduction

Organizing information requires the use of taxonomy, which is crucial for effective content management and content search. Taxonomy is the process of classifying and organizing information through a structured set of words intended for browsing. Maintaining consistent definitions and organization of concepts is essential for ensuring and preserving the reliability of information classification quality as it plays a vital role in improving the retrieval speed and classification of knowledge. Originally used in biology to classify biological specimens, taxonomy is now utilized in various fields such as psychology and information technology, where is particularly helpful for content management and information architecture. A taxonomic relationship is defined as a kind of hierarchical relation that encloses two concepts. One of which is a broad or general category, while the other is a more specific or narrow subcategory. See Arslan and Cruz (2022) for more information on taxonomies.

Predicting the semantic relationship between words in a sentence is essential for Natural Language Processing (NLP) tasks. Deep neural language models accomplish outstanding results in multiple tasks involving semantics evaluation. The question posed by the shared task Presupposed Taxonomies: Evaluating Neural Network Semantics (PreTENS) is whether neural models can detect the taxonomic relationship between nouns, especially in scenarios where the pattern and/or the set of nouns in the sentence is previously unseen (Zamparelli et al., 2022). Sub-task 1 is a simpler classification task, while sub-task 2 is a more complex regression task. Both sub-tasks involve datasets in English, French and Italian. For each sub-task, teams are permitted three submissions. For each submission, the score is averaged over the three languages. The highest score from the three submissions is reported.

We propose a series of models based on pre-trained language models. We enhance the provided datasets using state-of-the-art data augmentation tools, and further increase the dataset size by employing translations. The aim of both steps is to create slightly modified versions of the sentences, such that the model can learn alternative forms of nouns and patterns.

For the classification task (sub-task 1), we obtained the 3rd place, with an $\text{F1}_{\text{Binary}}$ score of 91.25% averaged over the three languages. For the regression task (sub-task 2), we obtained the 5th place, with a Spearman's correlation coefficient ρ of 0.221 averaged over the three languages. Sub-task 2 is markedly more difficult than sub-task 1 due to sentences that can be ambiguous, such as *I like dogs, but not chihuahuas*; some humans will judge this sentence as acceptable, while some will not. We attempt to solve both tasks by employing

data augmentation techniques in order to help the models understand variations in text. Our main contributions are:

1. We devise a special development-validation split to emulate the real situation in which the model must face new words and patterns.

2. We combine various data augmentation tools to allow the models to learn from various versions of the training dataset.

In Section 5.1 we present the task details and some of the related work that was done previously. In Section 5.2 we motivate our choice of models. The experiments we performed are in Section 5.3. Results and conclusions are presented in Sections 5.4 and 5.5 [1].

Related Work

For the present task, we are provided with a list of sentences following a set of *patterns*, all of which have two slots for noun phrases. One such sentence might be: *'I don't like beer, a special kind of drink'*. The pattern corresponding to this sentence would be: *'I don't like* [blank]*, a special kind of* [blank]*'*. Sentences are labeled according to whether the taxonomic relation between the two nouns makes sense. In sub-task 1, labels are binary; a sentence such as that shown above has a label of 1, while this sentence would have a label of 0: *'I like huskies, and dogs too'*. In sub-task 2, labels are continuous, ranging from 1 to 7; these scores are based on a seven-point Likert scale, judged by humans via crowdsourcing. The same dataset is presented in English, Italian and French. For sub-task 1, the training and test sets consist of 5 838 and 14 556 sentences, respectively; for sub-task 2, the training and test sets consist of 524 and 1 009 sentences, respectively. There are two challenges to this dataset:

1. The test dataset is much bigger than the training dataset.

2. There are unseen patterns and noun pairs in the test set.

The combination of these hampers the ability of machine learning (ML) models trained on the training set to generalize well to the test set. Indeed, that is the aim of this task: to evaluate the ability of language models to generalize to new data when it comes to inferring taxonomies.

[1] Our implementation of UU-Tax is publicly available at `https://github.com/IS5882/UU-TAX`.

One way to conceptualize the PreTENS task is to reformulate it as a taxonomy extraction task with pattern classification and distributed word representations. For a given sentence, extract the noun pair and the pattern from the sentence, and then determine if the taxonomic relation between the nouns matches the relations allowed by the pattern. This formulation is motivated by previous work in taxonomy construction that relied on various approaches ranging from pattern-based methods and syntactic features to word embeddings (Huang et al., 2019; Luu et al., 2016; Roller et al., 2018). As promising as this approach sounds for PreTENS, it involves manual labeling of the noun-pair taxonomic relations in the training set, as we are not allowed to use resources such as WordNet (Fellbaum, 1998) or BabelNet (Navigli and Ponzetto, 2012).

A different approach is to tackle PreTENS as a cross-over task between extraction of lexico-semantic relations and commonsense validation. There have been SemEval tasks to extract and identify taxonomic relationships between given terms (SemEval-2016 task 13) (Bordea et al., 2016), and to validate sentences for commonsense (SemEval-2020 task 4, sub-task A) (Wang et al., 2020). The aim of the common-sense validation task is to identify which of two natural language statements with similar wordings makes sense.

In the SemEval-2016 task 13, approaches related to extracting hypernym-hyponym relations to construct a taxonomy involved both pattern-based methods and distributional methods. TAXI relied on extracting Hearst-style lexico-syntactic patterns by first crawling domain-specific corpora based on the terminology of the target domain and later using substring matching to extract candidate hypernym-hyponym relations (Panchenko et al., 2016). Another team designed a semi-supervised model based on the hypothesis that hypernyms may be induced by adding a vector offset to the corresponding hyponym word embedding (Pocostales, 2016).

Participants in the SemEval 2020 commonsense validation task had an advantage over PreTENS participants: they were allowed to integrate taxonomic information from external resources such as ConceptNet (Wang et al., 2020), which eased the process of fine-tuning the language models on the downstream task. As an example, the CN-HIT-IT.NLP team (Zhang et al., 2020) and ECNU-SenseMaker (Zhao et al., 2020) both used a variant of K-BERT (Liu et al., 2020b) with additional data; the former injects relevant triples from ConceptNet to the language model, while the later also uses ConceptNet's unstructured text to pre-train the language model. Other systems relied on ensemble models consisting of different language models such as RoBERTa and XLNet (Pai, 2020; Altiti et al., 2020).

In Section 5.2 we outline the architectures chosen to tackle the two subtasks of PreTENS. We draw on previous work, as outlined above, and provide

novel combinations of datasets and algorithms to improve the performance of out-of-the box language models.

5.2 System Description

The systems we propose for both PreTENS sub-tasks are based on language models. In sub-task 1 we use the ELECTRA (Efficiently Learning an Encoder that Classifies Token Replacements Accurately) transformer (Clark et al., 2020), while in sub-task 2 we employ USE (Universal Sentence Encoder) (Yang et al., 2020b).

Sub-task 1: Classification

In the first sub-task —binary classification— we were required to assign an acceptability label for each sentence in the three languages English, French and Italian. Of the 20 394 sentences that were provided for sub-task 1, only 5 838 sentences (28.61%) were available for training. This split causes the model to be likely to encounter unknown data formats at testing time. This is a pivotal challenge in PreTENS, as the robustness and generalization of language models is an open challenge and cannot be guaranteed (Tu et al., 2020; Ramesh Kashyap et al., 2021). In our experiments we found that every language model we used (BERT, RoBERTa, XLNet, and ELECTRA) failed to generalize well to unseen datasets, even though all of them are pre-trained on large amounts of data. To address this challenge, we built our models based on data augmentation.

While designing our model, we split the provided training data into a development set (30%) and a validation set (70%), to emulate the train-test split sizes. We deliberately leave several patterns out of the development set, including, for example: *'I like* [blank]*, and more specifically* [blank]*'*. We choose these so-called *complex patterns* because, during exploratory experiments, we found that pre-trained models had trouble with them. For example, out of the 820 instances of the aforementioned pattern in the training dataset, 750 instances were misclassified by one of the early instances of our model; this includes sentences where the noun pair was included in other sentences in the training data. We thus remove complex patterns from the training data, to simulate a situation in which new unseen and difficult patterns are found in the test set.

Transformer language models like BERT (Devlin et al., 2019b) are pre-trained on two tasks: Masked Language Modelling (MLM) and Next Sentence

Prediction (NSP). However, in subsequent models such as RoBERTa, training on NSP was proven to be unnecessary; these models are thus pre-trained solely on MLM. ELECTRA further enhanced MLM performance while utilizing notably less computing resources for the pre-training stage. The pre-training task in ELECTRA is built on discovering replaced tokens in the input sequence; to achieve this, ELECTRA deploys two transformer models: a generator and a discriminator, where the generator is trained to substitute input tokens with credible alternatives and a discriminator to predict the presence or absence of substitution. This setting is similar to Generative Adversarial Networks (GANs) (Goodfellow et al., 2014), with a key difference that the generator does not attempt to trick the discriminator, making ELECTRA non-adversarial. In ELECTRA, the generator parameters are only adjusted during the pre-training phase. Fine tuning on downstream tasks only modifies the discriminator parameters (Clark et al., 2020).

Multi-stage fine-tuning has proven its effectiveness on the robustness and generalization of models (Kocijan et al., 2019; Li and Rudzicz, 2021). We perform a 2-stage fine-tuning; Figure 5.1 portrays our model work-flow. In the first stage, we use the NLPAug tool (Ma, 2019) to generate new sentences by making modifications to existing sentences based on contextualized word embeddings. There are several actions for the NLPAug tool; we utilize the 'Insertion' and 'Substitution' operations. The 'Insertion' operation picks a random position in the sentence, and then inserts at that position the word that best fits the local context. Meanwhile, the 'Substitution' operation replaces a word in a given sentence by the most appropriate alternative for that word. In both operations, the word choice is given by contextualized word embeddings, as will be explained in Section 5.3. To avoid drifting away from the original sentence, in both operations we limit the number of insertions and substitutions to two. Because 'Substitution' in NLPAug might turn an incorrect sentence into a correct one, we only carry out 'Substitution' on sentences labeled 1. An example of the output of the NLPAug tool is shown in Figure 5.4 in Section 5.6.

The second stage of fine-tuning also involves data augmentation, using translation. For each language l, we translate the datasets of the other two languages into l. For example, as seen in Figure 5.1, when working on the English model, we translate the Italian and French datasets to English, and perform the second fine-tuning stage on the translated data along with the original data. We use the Google Translate API for all translations [2].

[2]Only 15% of the translated sentences using Google Translate API were duplicates of the original sentence.

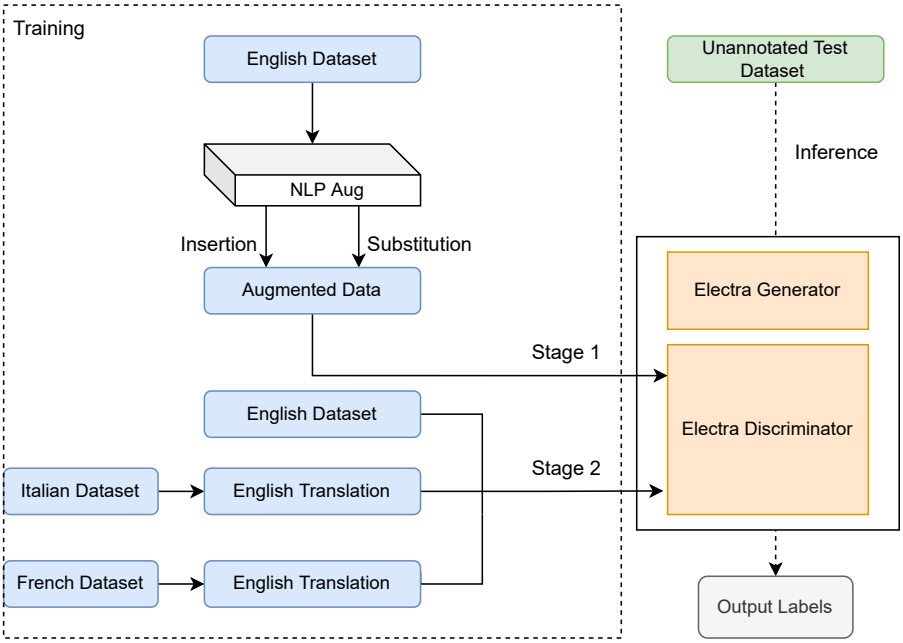

Figure 5.1: Sub-task 1: The English version of the proposed two-stage fine-tuning model (UU-Tax). In the French version, the Italian and English data are translated to French, and the NLPAug tool is employed on the provided French training set. Likewise in the Italian version.

Sub-task 2: Regression

In sub-task 2 —regression— we are required to determine the level of acceptability of sentences on a seven-point Likert scale. Our initial attempt in sub-task 2 resembles the efforts made in the first sub-task by relying on pre-trained language models. However, our first submission, which relies on fine-tuning multi-lingual BERT (Devlin et al., 2019b) with translation as data augmentation, did not perform well; more elaboration on this in Section 5.4. As a result, we opt for a simpler yet more effective model using Universal Sentence Encoder (USE) (Yang et al., 2020b) followed by a regressor. USE is based on two encoder models and deep averaging networks; both are equipped to generate a 512-dimension sentence embedding from a given textual input, where embeddings for words and bi-grams are averaged together and then passed

Table 5.1: Sub-task 1: Hyper-parameters values for training the ELECTRA model. The number of epochs and the batch size were determined by validation.

Hyper-parameter	**Value**
Epochs	4
Batch Size	8
Stage 1 Learning Rate	3×10^{-5}
Stage 2 Learning Rate	4×10^{-5}
Optimizer	AdamW

as input to a deep neural network that processes and outputs the sentence embeddings.

5.3 Experimental Set-up

Sub-task 1: Classification

We implement our submitted models using SimpleTransformers[3]. All models are trained for 4 epochs with a batch size of 8; these values were determined by validation, as we explain below. The model is optimized using AdamW (Loshchilov and Hutter, 2019) and a linear decay learning rate schedule. The learning rate is a key aspect of the performance of a trained model. A large learning rate results in quick model convergence; however, if the learning rate is too large, it will lead to drastic updates that will trigger divergent behaviour, while training a model with a too-small learning rate might lead to an underfitted model that gets stuck in local minima (Bengio, 2012). In our two-stage model, the first stage has a lower learning rate of 3×10^{-5} as opposed to the 4×10^{-5} assigned in the second stage, which contains the PreTENS training data; this is because we want the model to learn more from the real training data than from the NLPAug-edited data. A summary of the model hyper-parameters is given in Table 5.1. All the hyper-parameters are tuned based on the F1 score on the validation set. The same hyper-parameters are utilized for all three languages—English, French and Italian.

For data augmentation with NLPAug, $\text{BERT}_{\text{base}}$ is employed to obtain the contextual word embeddings for both 'Insertion' and 'Substitution' operations.

[3]https://github.com/ThilinaRajapakse/simpletransformers

Sub-task 2: Regression

For the three languages English, French and Italian we deploy multi-lingual $\text{USE}_{\text{Large}}$ as it yields better performance than mono-lingual USE for the three languages. USE is employed through its Tensor-Flow hub module[4]. We experiment with four different regressors: Linear Regression (LR) (Montgomery et al., 2021), K-Nearest Neighbors Regressor (KNR) (Kramer, 2013), Decision Tree (DT) (Myles et al., 2004), and Support Vector Regressor (SVR) (Awad and Khanna, 2015). We use the Scikit-Learn (Pedregosa et al., 2011) library for the implementation of the regressors. All regressors are utilized with their default parameters except for SVR epsilon ε . To define a higher margin of tolerance where no penalty is given to errors we set ε to 0.2 rather than the default value of 0.1.

Evaluation measures

Sub-task 1 is evaluated using the Binary-averaged F1 score ($\text{F1}_{\text{Binary}}$) for each language, while the global rank score is calculated as the average of the $\text{F1}_{\text{Binary}}$ for all three languages. Sub-task 2 is evaluated using Spearman's rank correlation coefficient (ρ) for each language, with the global rank given by the average of the coefficients for all languages.

5.4 Results and Evaluation

In this section, we analyze the performance of our submitted models in both sub-tasks. We further discuss other notable experiments that were carried out.

Sub-task 1: Classification

Results of the submitted models for English, French, and Italian are shown in Table 5.2. Out of 21 teams, we were officially ranked 3rd in sub-task 1, achieving a global score of 91.25%, only 1.06, and 2.92 percentage points short of the 2nd and 1st places, respectively. In the next few sections, we explain how our experimentation led us to the model we chose: the two-stage fine-tuning using ELECTRA with data augmentation (UU-Tax).

Table 5.2: Sub-task 1: UU-Tax submission results using a two-stage fine-tuned ELECTRA model.

Language	**Results**		
	Recall	**Precision**	**F1**$_{\text{Binary}}$
English	95.26 %	90.54 %	92.84 %
French	93.14 %	85.83 %	89.34 %
Italian	90.47 %	92.69 %	91.57 %
Average			91.25%

Table 5.3: Sub-task 1: Comparison of the different experiments carried out on the English Language.

Model	**Results**		
	Recall	**Precision**	**F1**$_{\text{Binary}}$
Baseline (TF-IDF + SVR)	85.64 %	64.19 %	73.38 %
Multi-task fine-tuning	**95.82** %	83.45 %	89.21 %
Data-enriched fine-tuning (BERT + Bi-LSTM)	86.70 %	89.79 %	88.22 %
UU-Tax (two-stage ELECTRA)	95.26 %	**90.54 %**	**92.84 %**

Experiments

Baseline. The PreTENS organizers proposed a baseline algorithm that trains an SVM classifier with features generated by TF-IDF with n-grams ($n = 3$). Results of the baseline model are reported in Table 5.3.

Multi-task fine-tuning. We experimented with several models on the English dataset. We tried a multi-task approach that involves further fine-tuning on related data-rich supervised tasks. In our case, it was the 'common sense validation' task, as it is highly correlated to PreTENS as previously mentioned in Section 5.1. We used the dataset from SemEval-2020 Common Sense Validation sub-task A (Wang et al., 2020) and modified the sentence label to 1 if it is a valid sentence and 0 otherwise. We then fine-tuned our ELECTRA model in the first stage using this data; the second stage of fine-tuning was carried out using the augmented data from NLPAug and the provided training data. Multi-task fine-tuning has proven its effectiveness across a variety of tasks (Mahabadi et al., 2021). This model achieved an F1$_{\text{Binary}}$ of 89.09%, which demonstrates the effect of information sharing between the different tasks, par-

[4]https://tfhub.dev/google/universal-sentence-encoder-multilingual-large/3

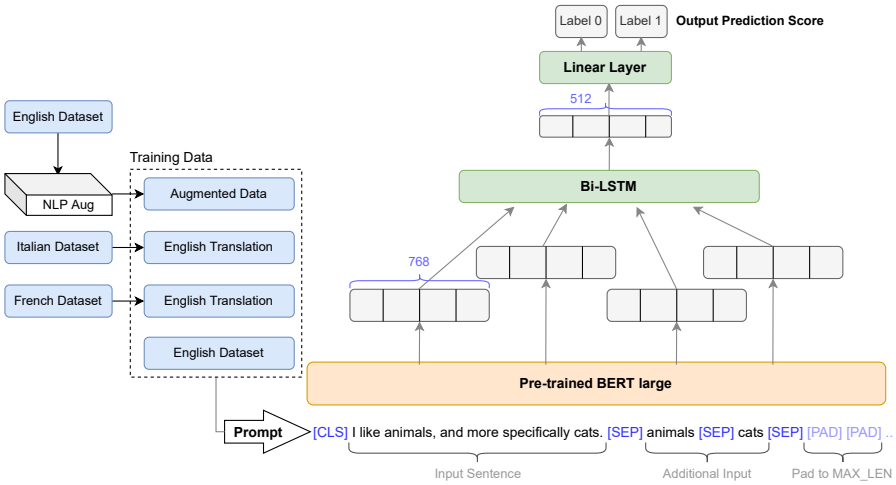

Figure 5.2: Sub-task 1: data-enriched fine-tuning model that employs an Bi-LSTM network on the top of pre-trained BERT. This model was used during the experimentation phase.

ticularly in cases when the downstream task is of a limited size. Nevertheless, multi-task fine-tuning suffers from several shortcomings including catastrophic forgetting, over-fitting in low-resource tasks and under-fitting in high-resource tasks (Mahabadi et al., 2021). For this reason, we did not move forward with this approach.

Data-enriched fine-tuning. As an alternative, we developed a data-enriched fine-tuning model that employed a pre-trained BERT model with an additional Bidirectional Long Short Term Memory (Bi-LSTM) layer on top. In addition to the input sentence, we concatenated the two nominal arguments to the given input. To extract the two nouns from the sentences, we leveraged the fact that nouns in this dataset tend to have very low document frequencies (DF), and classified any word with DF less than 5% as a noun. The final prompt of the input was as follows: $[\text{CLS}]Sentence[\text{SEP}]Noun\,1[\text{SEP}]Noun\,2[\text{SEP}]$ Similar to the aforementioned models, we also input to the model the augmented data generated from NLPAug. This model was implemented with PyTorch using the Hugging Face[5] Transformers library (Wolf et al., 2020). Figure 5.2 depicts

[5]https://huggingface.co/

the data-enriched fine-tuning model. The model's performance resembles that of the multi-task fine-tuning model by achieving an $F1_{Binary}$ of 89.04%.

As shown in Table 5.3, our submitted two-stage fine-tuning ELECTRA model (UU-Tax) achieved the highest results amongst all models, by a margin of 3.63% and 4.62% between both multi-task learning model and data-enriched fine-tuning model, respectively. We have almost 20% improvement compared to the baseline.

Ablation Study and Error Analysis

We conducted ablation experiments to evaluate the effect of data augmentation and our proposed two-stage fine-tuned ELECTRA model. The results of the analysis are presented in Table 5.4. We limit the ablation study and error analysis to the English dataset, as similar trends were observed in the French and Italian datasets [6].

Data augmentation effect. The need for data augmentation to generalize the model highly affects the performance of the pre-trained model. We perform two ablation analyses. In the first setting *(Ablation #1)*, we removed the translated dataset from the second stage, and our model was fine-tuned on data obtained from the NLPAug tool in the first stage and on the original training dataset in the second stage. The precision massively dropped by 11.42%. Similar behavior is observed in the second setting *(Ablation #2)*, when the NLPAug data is eliminated from our two-stage training, and the first stage is trained on the translated data instead, while in the second stage we fine-tuned using the original training data. This highlights the importance of our proposed dual augmentation using both NLPAug and translation to capture a wider range of perturbations to the original dataset.

Single-stage models' performance. To verify our two-stage fine-tuning approach, we evaluated it against a single-stage fine-tuning. This experiment was performed in two different settings; in the first *(Single-stage #1)* we trained on the originally provided data only, while in the second *(Single-stage #2)* setting we trained on the same data that was used in UU-Tax, which is obtained from NLPAug, translation, and the original training set. In both settings, we notice a drop in the F1 when comparing against UU-Tax. Nonetheless, we can observe that amongst the three experiments (UU-Tax, *Single-stage #1* and *Single-stage #2*) the highest recall of 96.26% is achieved in the (*Single stage*

[6]Results presented in Tables 5.3 and 5.4 may slightly vary due to fine-tuning instability of pre-trained language models (Mosbach et al., 2021).

Table 5.4: Sub-task 1: Results of various classification models trained during experimentation and ablation on the sub-task 1 dataset, using different combinations of input data obtained from NLPAug, translation (Trans) and the original training set provided (OT). Additional variations are single-stage versus two-stage models, and alternative pre-trained language models (LM). Recall (R), precision (P), and $F1_{Binary}$ (F1) are used as evaluation metrics. ✓ indicates which data is utilized in each fine-tuning stage, while - indicates that stage 2 is not applicable.

Model Name	Language Model	Stage 1			Stage 2			Results		
		NLPAug	Trans	OT	NLPAug	Trans	OT	Recall	Precision	F1-Binary
Ablation #1	ELECTRA	✓					✓	95.30 %	78.73 %	86.22 %
Ablation #2	ELECTRA		✓				✓	95.59 %	79.12 %	86.58 %
Single Stage #1	ELECTRA			✓	-	-	-	90.20 %	79.41 %	84.47 %
Single Stage #2	ELECTRA	✓	✓	✓	-	-	-	**96.26 %**	71.16 %	81.83 %
Two-Stage #1	BERT	✓				✓	✓	92.36 %	68.97 %	78.97 %
Two-Stage #2	RoBERTa	✓				✓	✓	93.93 %	78.24 %	85.37 %
UU-Tax	ELECTRA	✓				✓	✓	95.26 %	**90.54 %**	**92.84 %**

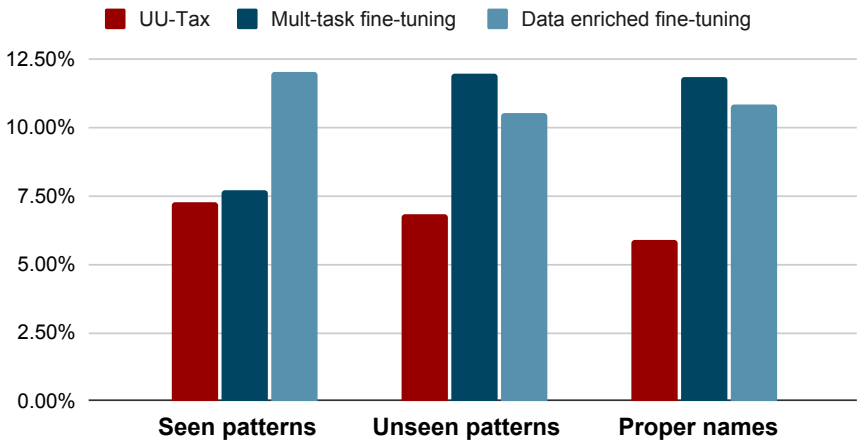

Figure 5.3: Sub-task 1: Percentage of incorrect predictions for all patterns in the test dataset, for the top three preforming models: UU-Tax, Multi-task fine-tuning and data-enriched fine-tuning.

#2) along with the lowest precision of 71.15%. Our interpretation of this finding is that in the *(Single stage #2*) experiment, the model over-predicted positives, causing the model to achieve a high recall and a relatively low precision. We attribute this behavior to two causes. First, the unbalanced ratio that NLPAug 'Substitution' operation caused as previously explained in Section 5.2[7]. Second, in UU-Tax a higher learning rate is deployed in the second fine-tuning stage, making the model focus more on the original dataset than on the NLPAug data.

Experimenting with different language models. Additionally, we experimented with different pre-trained language models, namely BERT *(Two-stage #1)* and RoBERTa *(Two-stage #2)*. As seen in Table 5.4, ELECTRA outperforms both RoBERTa and BERT by 7.47% and 13.92%, respectively, of the F1 score, which illustrates the strong generalizability of ELECTRA. Our findings agree with (Anaby-Tavor et al., 2020; Kumar et al., 2020), who demonstrate that generative models are suitable for data augmentation.

[7]The NLPAug 'Substitution' dataset is composed of 5568 instances all labeled '1', making 67.98% of the NLPAug data to have a '1' label.

Table 5.5: Sub-task 2: UU-Tax submission results that achieved the highest score averaged over the three languages, out of the three submissions. ρ is Spearman's rank correlation coefficient.

Language	**Model**	**Rho (ρ)**
English	USE + SVR	0.478
French	USE + DT	-0.059
Italian	USE + LR	0.246
Average		0.221

Table 5.6: Sub-task 2: Rho (ρ) scores of different regression models that we experimented. Models that were part of the global score are marked with an * . Baseline is TF-IDF + SVR; BERT is multilingual.

Language	**Model**						
	Baseline	**BERT**	**BERT + Trans**	**USE + LR**	**USE + KNR**	**USE + DT**	**USE + SVR**
English	0.247	-0.068	-0.027	-0.175	0.235	0.118	**0.478***
French	**0.230**	-0.075	-0.027	0.207	0.103	-0.059*	0.030
Italian	**0.370**	0.047	0.150	0.246*	0.081	0.171	0.137

Error Analysis. By manually inspecting the wrong predictions generated by our proposed top three performing models (UU-Tax, multi-task fine-tuning, and data-enriched fine-tuning) we can observe that UU-Tax achieves the smallest percentage of incorrect predictions on both seen and unseen patterns, as observed in Figure 5.3. This shows that the proposed two-stage fine-tuning (UU-Tax) can learn better and generalize better than multi-task fine-tuning and data-enriched fine-tuning. In addition, we also noticed that proper names were the cause of many misclassifications. One possible mitigation to overcome this error is to create an improved model to envision proper names appearing in a sentence as hyponyms of the preceding or the subsequent noun appearing in the same sentence.

Sub-task 2: Regression

As explained in Section 5.2, USE was employed for all three languages to obtain pre-trained word embeddings; we used SVR, DT, and LR regressors for English, French and Italian, respectively. We came in 5th in sub-task 2 out of 17 teams by achieving a global average of 0.221. It is worth noting that we had a better performing French-language model in the first submission than in our top submission. The experiments we performed for sub-task 2 are discussed in

Section 5.4. The ρ coefficients for the three languages in our best submission are reported in Table 5.5.

Experiments

Table 5.6 shows the results of our submitted models along with other experiments that we carried out using different regressors as explained in Section 5.3. In addition, we also experimented using multi-lingual BERT in two different settings; once with only fine-tuning on the provided dataset of the three languages and in the other setting, we augmented the provided training data with translation as in the translation process in sub-task 1.

In English our submitted USE + SVR model achieved the highest ρ score of 0.478 amongst all other models, surpassing the baseline by 94%. Although in the French version our final submitted model was, unfortunately, the model with the lowest score, we were able to achieve the highest score of 0.207 using LR, less than the baseline approach by $\Delta\rho = 0.023$. While in Italian, our submitted model was our highest rho score achieved of 0.246 which is $\Delta\rho = 0.123$ lower than the baseline. We infer from the fact that our model performed badly on French and Italian that USE is better optimized for English language.

Ablation Study and Error Analysis

Pre-trained language models did not perform well. We attribute this to the very limited training size of sub-task 2: only four different patterns made up the training data. The deployment of data augmentation—translation—to multi-lingual BERT was able to improve the performance on all three languages by more than 50%, which confirms our hypothesis that the limited pattern in the provided training set highly affected the performance of the pre-trained language model. This is supported by a similar trend when experimenting with different language models. Since this is a regression task, we were not able to use the NLPAug tool as the assigned score might be inaccurate after the substitution and insertion operations.

There is no consistently best performing classical ML algorithm: unlike for Italian and French, LR did not perform well on the English dataset, and SVR outperformed all other regressors on the English version. Interestingly, we see a consistent pattern across the French and Italian versions, showing that the LR regressor works best; we attribute this to the lexical and grammatical similarity between the French and Italian languages.

5.5 Conclusion

The limited size of the training dataset as compared to the test set made it impossible to train neural networks directly on the task. As a result, we took advantage of pre-trained language models. Nonetheless, the robustness of language models is highly affected by the size and variance of the downstream task data available for fine-tuning, which causes the language model to fail to generalize. Hereby, we relied upon data augmentation techniques using a two-stage fine-tuning process on ELECTRA. The first fine-tuning stage was carried out using an augmented version of the dataset, while in the second stage we used the translated versions of the provided PreTENS training data in addition to the original data. We ranked 3rd out of 21 teams in sub-task 1. For the second sub-task we proposed a simple model by training an SVR classifier with sentence embeddings obtained from USE; we ranked 5th out of 17 teams.

As an extension for future work, both sub-tasks could greatly benefit from adversarial training, which has proven its success across various NLP tasks in improving the model robustness and generalization (Liu et al., 2020c; Yoo and Qi, 2021).

5.6 Supplementary Materials

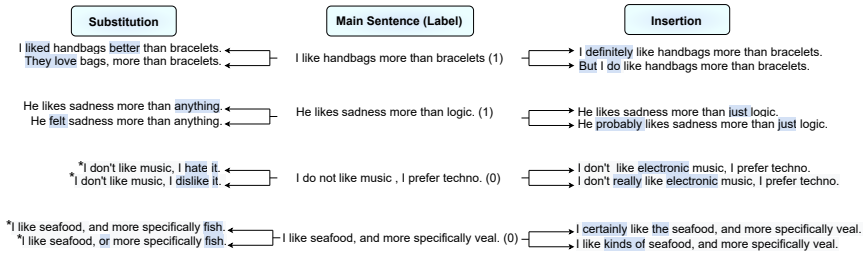

Figure 5.4: Sub-task 1: Example of the output generated by both, substitution and insertion operations of the NLPAug library. As explained in Section 5.2, for sentence with label 0, the substitution operation is not performed, this is indicated using an * in the figure.

6 End-to-end System for Knowledge Graph Construction Using Open Information Extraction

Instant analysis of cybersecurity reports is a fundamental challenge for security experts as an immeasurable amount of cyber information is generated on a daily basis, which necessitates automated information extraction tools to facilitate querying and retrieval of data. Hence, we present Open-CyKG: an Open Cyber Threat Intelligence (CTI) Knowledge Graph (KG) framework that is constructed using an attention-based neural Open Information Extraction (OIE) model to extract valuable cyber threat information from unstructured Advanced Persistent Threat (APT) reports. More specifically, we first identify relevant entities by developing a neural cybersecurity Named Entity Recognizer (NER) that aids in labeling relation triples generated by the OIE model. Afterwards, the extracted structured data is canonicalized to build the KG by employing fusion techniques using word embeddings. As a result, security professionals can execute queries to retrieve valuable information from the Open-CyKG framework. Experimental results demonstrate that our proposed components that build up Open-CyKG outperform state-of-the-art models.

This work was originally published as:
Sarhan, I. and Spruit, M. (2021). Open-cykg: An open cyber threat intelligence knowledge graph. *Knowledge-Based Systems*, 233:107524

6.1 Introduction

Cyber threats are developing at a rapid pace, which is driving security analysts to dynamically utilize various Natural Language Processing (NLP) techniques as means to defend, identify, analyze, and possibly mitigate various cybersecurity attacks. These include text memorization (Russo et al., 2019), information extraction (Gasmi et al., 2019; Jones et al., 2015) and Named Entity Recognition (NER) (Georgescu et al., 2021; Bridges et al., 2013). In order to understand the means and the consequence of different cyber-attacks, security professionals rely on previous reports, such as security bulletins or online reports, to get a better grasp of the threat at hand. Unfortunately, such reports are often stored in an unstructured manner, making efficient information retrieval even more challenging.

Currently, existing information extraction systems lack two essential components. First, a methodology that is capable of extracting valuable information that does not necessitate either a pre-defined set of relations or an existing ontology (Muhammad et al., 2020), limiting extraction to a specified set of information, thus increasing the probability of missing out on vital knowledge. Second, a data structure that supports storing extracted data efficiently to allow successful information retrieval and knowledge understanding. The absence of this kind of data structure will prevent security analysts from fully leveraging the extracted information. In this chapter, we introduce *Open-CyKG*: an open Cyber Threat Intelligence (CTI)[1] Knowledge Graph (KG) constructed from Open Information Extraction (OIE) triples.

Open-CyKG is a framework that is capable of efficiently extracting valuable information from unstructured Advanced Persistent Threat (APT) reports and representing the retrieved data in a KG that offers efficient querying and retrieval of threat-related information. Open-CyKG is made up of two main components as shown in Figure 6.1. First, an attention-based OIE architecture for extracting domain-independent relational triples from unstructured data. Second, a NER model for automatic labeling of cybersecurity terms. More precisely, we start by extracting structural relation tuples from APT reports using OIE, which are later populated in the KG with the help of the NER task.

Attention mechanisms have had notable success in several deep learning tasks (Vaswani et al., 2017; Gao et al., 2018; Liu et al., 2015). The first building

[1] CTI is the outcome of threat information once it has been compiled and analyzed to provide actionable advice regarding previously known or emerging threats that helps with the mitigation process (McMillan, 2013)

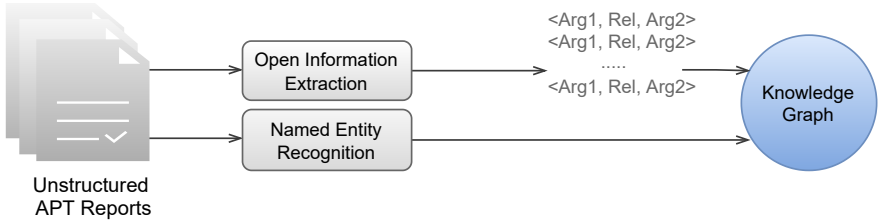

Figure 6.1: Main building components of the proposed Open-CyKG framework. A more detailed version is shown in Figure 6.2.

block is an attention-based OIE. We propose a novel attention mechanism that emphasizes the syntactic and semantic features of a given sentence in a way that words are assigned different weights based on their level of contribution to a sentence. We demonstrate that an attention-based approach improves the process of identifying semantic relations. The extracted tuples are composed of a predicate and a set of attributes in the form of $\ll$ *Argument 1, Relation, Argument 2* $\gg$.

The second component in Open-CyKG is NER, which acts as both a standalone NLP application and a pre-processing phase for several NLP tasks including, information extraction, question answering, and KG construction (Yadav and Bethard, 2018; Piplai et al., 2020). It has been significantly researched in different domains, but only a few studies have targeted the cybersecurity domain. We demonstrate the importance of a NER module in KG construction and refinement.

One of the significant challenges faced during the building process of a KG is data redundancy and ambiguity. Consequently, to overcome this challenge, we employ refinement and canonicalization techniques to fuse information in the KG based on their contextualized word embeddings by using hierarchical agglomerative clustering for entity grouping. Various empirical analyses to validate the components of Open-CyKG were carried out, demonstrating that OIE can highly support the development of knowledge bases. We address the above challenges by proposing a novel and open cybersecurity KG model. The contributions presented in this work involving different stages of Open-CyKG are as follows:

- We contribute the first OIE-based KG in the cybersecurity domain that does not limit extractions to a pre-specified set of information. Our model integrates OIE and NER with KG fusion techniques to produce an effective open cyber threat intelligence KG model: Open-CyKG.

- We introduce an attention-based sequence-to-sequence OIE model that outperforms state-of-the-art network architectures, hereby demonstrating its effectiveness in information extraction tasks.

- We develop a cybersecurity NER model to label prominent words in this domain that achieves notable results when compared against several baselines and state-of-the-art models.

- We conduct a refinement and fusion process in Open-CyKG, which uses the generated NER labels and contextualized word embeddings to further enhance the quality of the retrieved queries.

- We show that once Open-CyKG is created, information retrieval can be performed efficiently using two sample queries.

The remainder of this chapter is structured as follows. Section 6.2 reviews previous work in OIE, NER, and KG, followed by our proposed framework Open-CyKG in Section 6.3, while Section 6.4 presents results and evaluation of Open-CyKG. Finally, Section 6.5 concludes the chapter along with future work discussion. [2]

6.2 Related Work

In this section, we review previous work performed on Open Information Extraction (OIE), Named Entity Recognition (NER), and Knowledge Graphs (KGs) in the literature underlining previous work on the aforementioned tasks in the field of cybersecurity.

Open Information Extraction State-of-the-Art

OIE methodologies can be classified into three main categories as shown in Chapter 2: machine-learning classifier approaches, hand-crafted rules approaches, and neural network approaches. The first two approaches can be further classified into two categories, either deploying shallow syntactic analyses or dependency parsing techniques. In this section, we focus on neural network approaches and previous work in the cybersecurity domain.

[2] Our implementation of Open-CyKG is publicly available at `https://github.com/IS5882/Open-CyKG`

Neural Network Approaches

Deep neural network approaches have proven their reliability and success on a wide range of NLP tasks and recently made their way towards OIE systems as an alternative to feature-based methods, which are considered both time and effort-consuming to correctly capture entities and linguistic features. In 2018, Cui et al. (2018) developed a Recurrent Neural Network (RNN) encoder-decoder OIE framework that is constructed from a 3-layer Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997). By collecting training data from high-confidence state-of-the-art OIE systems, a variable-length sequence is inputted to the encoder. Subsequently, the resulting compressed representation vector is used by the decoder to produce the output sequence. Additionally, in the same year, Stanovsky et al. (2018) proposed a supervised OIE paradigm that utilizes a Bidirectional LSTM (Bi-LSTM) transducer to train the neural network for tuples extraction. The authors also validate that OIE can immensely benefit from an automatic question-answering-semantic role labeling extractor. In Chapters 3 and 4, a Bidirectional Gated Recurrent Units (Bi-GRU) OIE model was introduced that leverages contextualized word embeddings.

SpanOIE (Zhan and Zhao, 2020) is the first span OIE model that adapts the same idea of modified span selection that is employed in co-reference resolution, syntactic parsing, and semantic role labeling. The span model's key benefit is visible when applied to token-based sequence labeling models in which span-level syntactic information can be adequately exploited. The authors emphasize that features of span level support better extraction quality. Cabral et al. introduced CrossOIE (Cabral et al., 2020), a multilingual OIE model that deploys convolution neural networks that support English, Spanish and Portuguese extractions. The cross-language OIE model employs a binary classifier that generates training features from cross-language embeddings, with this highlighting the importance of developing cross-lingual OIE systems as research is more focused on the English language only.

Information Extraction in Cybersecurity

To the best of our knowledge, no previous work has been performed that specifically explores OIE in the field of cybersecurity. However, several other information extraction techniques targeting the cybersecurity domain were recently proposed. In 2019, Georgescu et al. (2021) proposed an LSTM-based Relation Extraction (RE) system for mining predefined cybersecurity entities from text. Contrasting to OIE, in RE task relations must be predetermined

prior to extraction. The authors focus primarily on identifying relations that relate to vulnerabilities that link software with vendors or specific files. Another RE framework was introduced by Jones et al. (2015) that applies a bootstrapping pattern-based approach for tuple extraction. Their model integrates active learning components that query the user to supply precise input into the system. Similar to Gasmi's RE model, Jones et al. (2015) use predefined relations that correspond to attributes of vulnerability and software which are derived from a cybersecurity ontology, both RE models aim for a better understanding of vulnerability-related information.

Named Entity Recognition State-of-the-Art

NER is a well-researched topic in several different domains, where news and biomedical fields dominated the research in the NER task compared to other less-popular fields like cybersecurity. It is the task of identifying and locating named entities in unstructured text corpora and classifying them into a predetermined set of categories like location, person, organization, date, and time expressions. Over the last few years, Neural Network (NN) techniques have taken over the lead in NER systems as they minimize the need for human effort in constructing features and rules necessary for achieving a decent level of accuracy. In this section, we review the innovative NN approaches.

Neural Network Approaches

Deep learning systems require minimal feature engineering without essentially requiring lexicons or ontologies, thereby making them more domain-independent. Amongst the first NN-based NER systems introduced in 2008 is (Collobert and Weston, 2008), where a single Convolutional Neural Network (CNN) architecture was utilized to create a multi-tasking learning system that predicts named entities, POS tags, and semantically similar words. Additionally, the authors demonstrated that simultaneous task learning enhances the model's generalization. GRAM-CNN (Zhu et al., 2018) is another NER approach that uses CNN for biomedical entity extraction. The authors of (Zhu et al., 2018) used character embeddings instead of word embeddings to get a more informative representation of the words, where labels are predicted via a Conditional Random Field (CRF) layer.

RNNs paved their way to the NER task by either employing LSTM or GRU networks. CharNER (Kuru et al., 2016) is a character-level tagger that exploits stacked Bi-LSTM for encoding patterns; a decoder is then utilized to transform the generated character-level probability representation to word-

level tags. Opposing the character-level model, in 2016, Yang et al. (2016a) introduced a multi-tasking, language-independent NER model that concatenates both character-level and word-level features. To encode both of the aforementioned features, a hierarchical GRU is utilized before passing its output to a CRF layer for sequence tagging prediction.

Named Entity Recognition in Cybersecurity

As the demand for automatic text processing and information extraction relatively increased in all domains, NER found its way to the cybersecurity field. Nonetheless, from a technical perspective, NER systems introduced for the cybersecurity domain are highly comparable to the aforementioned systems.

(Bridges et al., 2013) implemented a maximum entropy model for labeling named entities on three different cybersecurity datasets: Microsoft Security Bulletin, National Vulnerability Database (NVD), and Metasploit Framework database, all of which were made publicly available by the authors. Average perception is trained on a fragment of the datasets while constantly monitoring successful entity classifications. Moreover, unigram and bigram features were also included.

Kim et al. (Kim et al., 2020) built a NER system using a deep Bi-LSTM-CRF neural network to automatically extract named entities of cyber threats. The key idea is to incorporate several features in their proposed model, mainly characters based on Bag-Of-Character (BOC) representations. Their predefined named entities comprise a diverse set of cybersecurity terms such as malware, hash, and Common Vulnerabilities and Exposures (CVE). Their model adapted character-level features, and additionally, GloVe (Pennington et al., 2014) was utilized to embed words.

In addition to the cybersecurity RE model proposed by (Georgescu et al., 2021) —- discussed in Section 6.2 —- they also introduced a NER system that exploits a Bi-LSTM-CRF neural network similar to the model proposed by (Kim et al., 2020) with the exception of adapting word-level features instead of character-level ones. Similar to (Bridges et al., 2013), the NVD dataset was employed for training and testing purposes. Another deep learning approach that combines Bi-GRU with CNN was designed by Simran et al. (2019). The Bi-GRU layer polishes the vectors prior to feeding them to the CNN layer, where features are fine-tuned before passing them to the CRF prediction layer.

Knowledge Graph Overview

With the arrival of the automatic information extraction and question-answering era, KGs[3] fulfilled the need to effectively mine structured knowledge from exhaustive texts. In this section, we start by briefly presenting the most popular KG applications and other NLP tasks that can benefit from KGs, followed by walking through different methods to build KGs and different canonicalization techniques. Finally, we review previous KG work in the cybersecurity domain.

Knowledge Graph Applications

The first KG was introduced by Google (Amit, 2012) in 2012, with the main objective of enhancing query results and further enriching the overall search experience of end-users. This was the start that ignited research in KG and the development of other KG-based applications. DBPedia (Lehmann et al., 2015) is a well-known multi-lingual KG project that permits users to retrieve information through semantic queries; data in DBPedia is mainly acquired from Wikipedia infoboxes. Freebase (Bollacker et al., 2008) is a collaborative knowledge base where community members compose the data, also described as *"an openly shared database of the world's Knowledge"*. It is worth noting that Freebase powered a part of Google's KG, however, it went offline in 2016 and was succeeded by Wikidata (Pellissier Tanon et al., 2016).

In addition to the aforementioned applications, KG also aided several NLP tasks, from information extraction (Hoffmann et al., 2011; Fei et al., 2020) and question answering (Bordes et al., 2014) to recommendation systems (Wang et al., 2019).

Knowledge Graph Construction and Canonicalization

There are several manners to construct a KG. It can be curated from existing knowledge bases like YAGO (Fabian et al., 2007) and Wikipedia, where the latter was mainly used in building DBPedia (Lehmann et al., 2015), or the KG can be populated and modified by users as in Freebase (Bollacker et al., 2008) and Wikidata (Pellissier Tanon et al., 2016). A third option is using information extraction techniques to obtain data from unstructured or semi-structured text to create the KG. As stated in (Bordes and Gabrilovich, 2014), whichever of the three methods is utilized to build the KG, it will never be entirely correct or complete. As is the case in DBPedia, although it has almost

[3]Knowledge Graph can be defined as a form of a data structure, which is composed of nodes and edges that are leveraged as a way to manage and illustrate information in such a manner that users can efficiently query and obtain data on a specific topic.

4.6 million entities, only half of them include fewer than five relations. Hence, KG canonicalization is required to overcome this challenge by employing fusion and refinement techniques to improve the overall quality of the KG, which might result in a trade-off between accuracy and coverage of the KG (Vashishth et al., 2018b).

In (Trisedya et al., 2019), an attribute character embedding that is formed on representation learning is created. The model uses the aforementioned embeddings to distinguish similarities between entities in a KG. Additionally, transitivity rules are applied to further enhance the attributes of an entity and assist in the entity-linking process. Another way to ensure that similar entities and relations in a KG lie in the same space is by using entity descriptors as deployed by Zhong et al. (2015). The alignment model produced by the authors of (Zhong et al., 2015) does not require dependencies on specific data sources. Therefore it can be integrated into any KG as long as the entities are identified. Entity linking is another way of canonicalization, by mapping entities in the text to existing entities in a KG as in (Radhakrishnan et al., 2018; Thawani et al., 2019). While entity disambiguation is deemed as a sub-task of entity linking, it is the process of linking the identified entity in a KG to a ground truth entity as in (Huang et al., 2015; Mulang' et al., 2020).

Knowledge Graphs in Cybersecurity

Following the same trend of the two aforementioned NLP tasks, OIE and NER, KG in the cybersecurity domain is one of the most under-researched domains compared to the more popular fields, such as news and biomedical domains. Narayanan et al. (2018) built a collaborative framework with the help of semantically rich knowledge representations. Their cognitive assistant system designated for early detection of cybersecurity attacks acquires vulnerability-related data from recently published threat intelligence reports from multiple sources such as online blogs and CVE reports, whereas information is later illustrated in a pre-constructed KG that is previously loaded with information such as early detected threats, attack patterns, and tools required to carry out an attack.

As an alternative to constructing a KG from data about vulnerabilities, Piplai et al. (2020) populate a cybersecurity KG from malware After Action Reports (AAR), as they enclose insightful analyses of cybersecurity incidents, hereby delivering reliable information to security analysts. As AARs provide crucial data about detection and mitigation techniques of attacks, they can also aid in dealing with new unidentified cybersecurity incidents by matching pattern similarities with a predefined incident. Additionally, to ease the ex-

traction phase, a traditional malware entity extractor that is based on Stanford NER (Finkel et al., 2005) was created that was trained on CVE and security blogs to label each word accordingly.

A further cybersecurity KG that is constantly maintained is SEPSES, introduced by (Kiesling et al., 2019). SEPSES encapsulates and relates essential information ranging from vulnerabilities to attack patterns and weaknesses. Data in the KG is populated from several sources, and amendments are instantly incorporated into the real world. For example, CVE data is continuously fed to their model and updated every two hours, which is valuable in capturing alerts caused by intrusion detection systems in parallel with providing updated vulnerability assessments.

Nevertheless, the aforementioned work in this section either depends on structured text to populate the KG or limits extractions to a predefined set of information. For the work of (Piplai et al., 2020), the authors employ RE, while authors of (Kiesling et al., 2019) completely rely on Apache Jena for the triple formulation of specific relations. This further demonstrates the strength of Open-CyKG, as it employs OIE, so it is not restricted to a predetermined set of relations or an ontology to extract information from cybersecurity reports.

6.3 Open-CyKG Framework

Our Open-CyKG framework is presented in Figure 6.2. The pipeline is composed of three main modules; a neural OIE system to extract relation triples from unstructured APT reports, a cybersecurity NER model that identifies and classifies each word according to a predefined set of labels, and a KG construction and fusion phase where extracted triples from the OIE phase are illustrated. The KG is constructed such that the extracted entities represent the KG nodes, and edges correspond to the extracted relations that couple the entities.

Neural OIE Model

Our OIE model is schematically presented in Figure 6.3. It is an upgrade on our previous OIE work described in Chapters 3 and 4. We tackle OIE as a sequence labeling task using the BIO (Beginning, Inside, and Outside) labeling scheme (Ramshaw and Marcus, 1999b) in such a way that the resulting outcome is a set of overlapping tuples for each sentence.

Due to the fact that RNNs have the capacity to store information in their hidden units, they are considered a suitable choice for handling sequential data

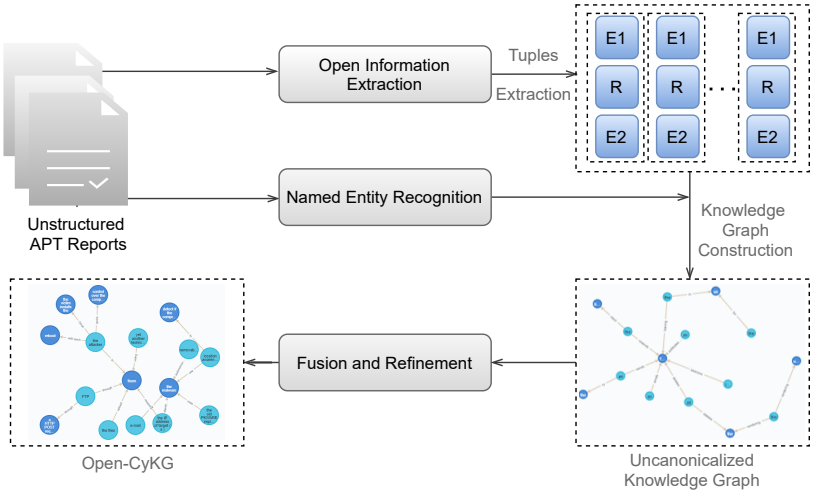

Figure 6.2: The Open-CyKG pipeline is composed of three primary building blocks: OIE, NER modules, as well as fusion and KG refinement techniques that result in a canonicalized open KG.

when compared to feed-forward artificial neural networks like CNNs, which also struggle to capture long-distance dependencies between words. Nevertheless, RNNs are harder to train with longer sequences owing to vanishing and exploding gradient descent complications, which leads the performance to degrade noticeably. As RNNs are trained by back-propagation through time, the further we back-propagate through several time steps, the smaller the gradient gets up until it vanishes or explodes. As a solution, extended versions of RNNs were introduced —LSTMs and GRUs— to mitigate vanishing gradient issues by deploying appropriate gates to permit the gradient to flow effectively while maintaining long-term dependencies (Pascanu et al., 2013; Hochreiter and Schmidhuber, 1997). GRUs are regarded as a less complex variation of LSTMs, both are built on the gating concept, where LSTM's architecture is compiled of three gates; input, output, and forget, while GRUs couple the input and forget gates into a single update gate. Our choice of deploying GRUs instead of LSTMs is also motivated by the fact that GRUs utilize fewer training parameters, resulting in quicker execution and training times in contrast to LSTMs.

In addition to embedding the word and its corresponding POS and passing them as input to our neural network, the feature vector is further enriched by

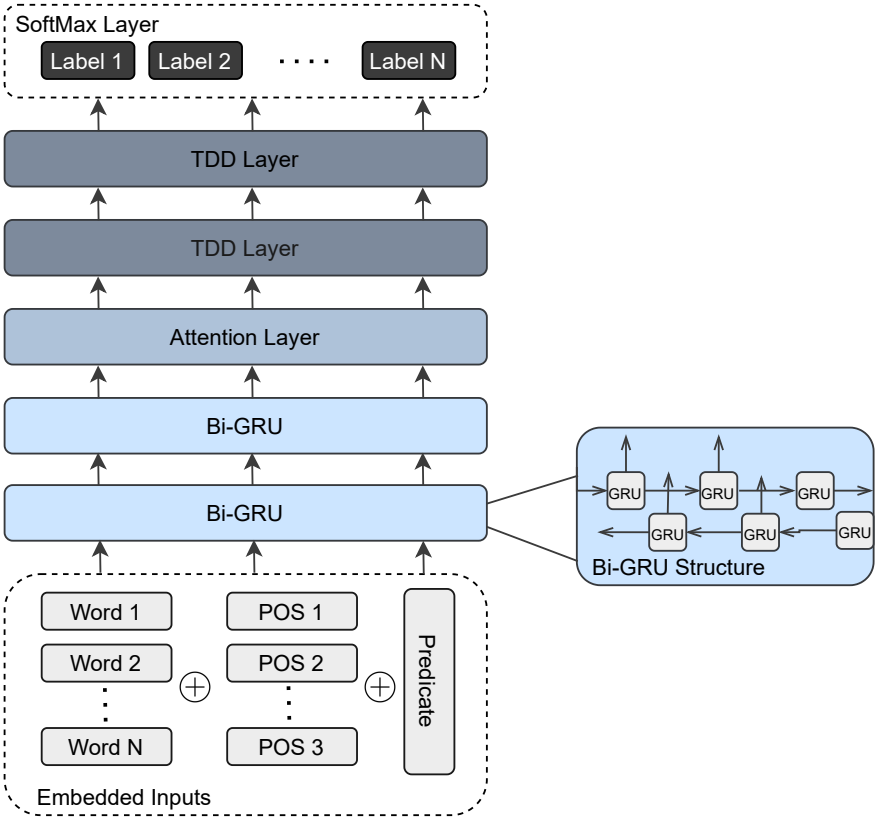

Figure 6.3: Our OIE model takes the concatenation of all inputs and passes the input to the two Bidirectional Gated Recurrent Units (Bi-GRU) layers, followed by an attention layer, two Time Distributed Dense (TDD) layers, and finally, a SoftMax layer for prediction.

passing the predicate of the phrase, as predicates are regarded as the building block of any sentence. The input feature vector ($F.V$) is represented in Equation 6.1.

$$F.V = (Emb(w) \oplus Emb(POS(w)) \oplus Emb(w_{Pred}))|w \in S \tag{6.1}$$

Where $\oplus$ represents the concatenation of the three inputs: word denoted as w, its corresponding POS obtained using the NLTK toolkit (Bird, 2006) $POS(w)$ and the predicate of the sentence w_{Pred}, where each word belongs to a sentence S. All the prior mentioned inputs are embedded Emb using contextualized word embeddings as discussed in Section 6.4.

The embedded feature vector is then passed to two Bi-GRU layers. As shown in Figure 6.3, Bi-GRU is composed of two GRUs operating in a reverse direction. The significance of utilizing a Bi-GRU rather than a single-direction GRU is that information is captured in both directions, forward and backward during each time step to perform sequence labeling efficiently.

The outputted tensor from the Bi-GRU layer is then passed through an attention layer (Bahdanau et al., 2015) that is based on an additive attention module. Most of the proposed OIE neural models discussed in Section 6.2 are formulated in a way such that all words have the same level of importance, however, it is essential to highlight that not all words in a sentence have an equivalent level of contribution in the OIE task. To address this issue, we employ the attention mechanism in our network to learn the varying significance of words in each phrase and aggregate the output to the following double-layered Time Distributed Dense (TDD) layer that employs a consistent dense layer to the passed tensor at each time step. As a final point, the tensor produced by the TDD layers is fed into a SoftMax layer, where the output is an individual probability distribution covering all possible tags.

Cybersecurity-NER

The task of classifying cybersecurity entities in our dataset resembles the efforts of prior research discussed in Section 6.2, however, with a differently designed neural network as illustrated in Figure 6.4. In a similar manner to our OIE approach, we formulate the NER task as a sequence labeling problem with BIO taggers, as it is considered the most suitable tagging module for NER, specifically in neural models employing CRF (Reimers and Gurevych, 2017a), where each word in the dataset is labeled according to a set of predefined entities based on its position.

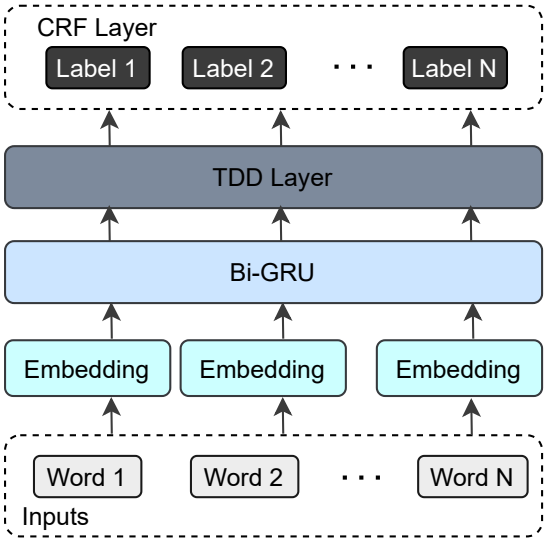

Figure 6.4: Our implemented cybersecurity NER neural model architecture is composed of four main layers; Embedding, Bidirectional Gated Recurrent Units (Bi-GRU), Time Distributed Dense (TDD), and Conditional Random Field (CRF) for label prediction.

Figure 6.5: An uncanonicalized sample of the created KG using Neo4J.

Initially, words are translated into their respective embeddings and are progressed directly to the following layer. Since long dependencies modeling in NER is essential, we also opted to deploy an RNN. To capture both backward and forward information, our proposed cybersecurity-NER deploys a Bi-GRU layer that outputs a tensor later passed to a TDD layer. Finally, a CRF prediction layer labels each word in our dataset by generating likelihood distributions over every available tag.

Knowledge Graph Construction and Canonicalization

To produce Open-CyKG, relation triples extracted from the OIE stage are processed and outlined in the KG as defined in Equation 6.2:

$$KG = \{(nh, e, nt) | nh, nt \in E, e \in R\} \tag{6.2}$$

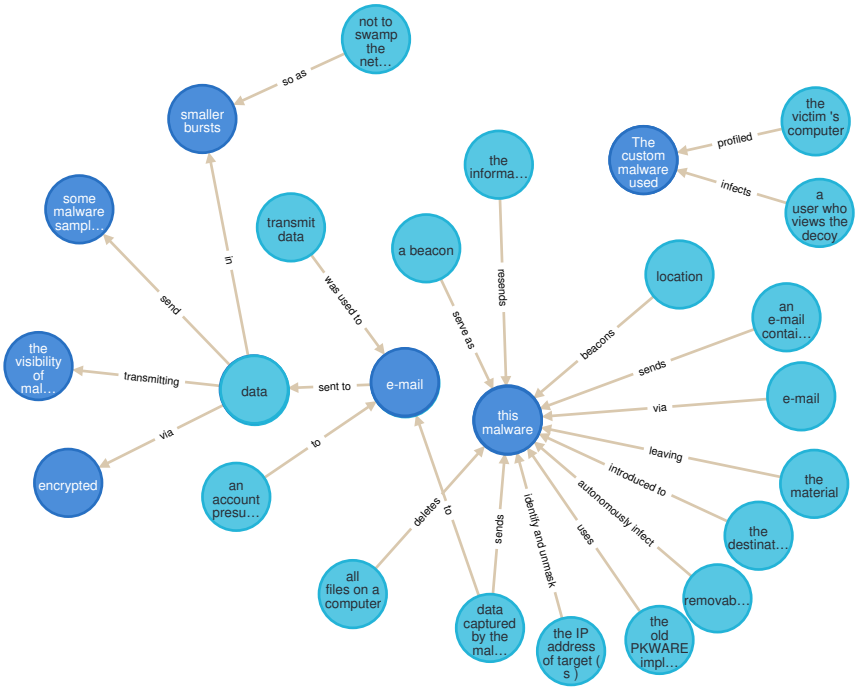

Figure 6.6: Canonicalized version of Figure 6.5 using Neo4J.

Where the set of the extracted OIE triples (nh, e, nt) are composed of a node head nh and a node tail nt in which both belong to the entities E, both nodes are linked together using an edge e that represents the relation R that lies between the two entities. Additionally, named entity tags are allocated to each node as a property. An uncanonicalized sample of the generated KG using Neo4J (Miller, 2013) is illustrated in Figure 6.5.

Several sources of information are used when constructing a KG, possibly prompting duplication. As a result, it is essential to apply refinement and fusion techniques to address this matter. The leading step is triple refinement: this two step-process involves removing redundant and vague information and entity blending, where identical entities are merged together after identifying and removing non-essential words to preserve only informative entities. The filtration task also involves eliminating uninformative triples generated from the OIE phase, in which all words forming the three extracted components are

not assigned any named entity labels from the cybersecurity NER phase.

Another common setback in the construction process that is not captured in the previous step is entity disambiguation, which can be perceived in two contradictory ways. The first ensures that an entity represents the same semantic concept to all its connected nodes, while the second unifies and merges entities that represent identical concepts. Entity disambiguation in KG is considered a research problem on its own that is out of the scope of this study. Nevertheless, we briefly attend to this issue by performing entity fusion using contextualized word embeddings to capture the semantics of entities. In our work, we experiment with several word embeddings discussed in detail in Section 6.4.

The potential of using word embeddings in Open-KG canonicalization to address ambiguity in the generated knowledge graph has been demonstrated by the works of (Vashishth et al., 2018b; Wu et al., 2020). By first averaging the generated word embeddings of all subjects in an entity, we cluster entities by carrying out Hierarchical Agglomerative Clustering (HAC) by employing cosine similarity as a distance metric. Our choice behind HAC is motivated by the fact that it does not require predefining the number of clusters in advance. Additionally, it supports complete linkage clustering, similar to the concept of farthest neighbor clustering, where initially, each average embedding is a cluster on its own, and in each step, the two clusters having the smallest maximum pairwise distance are merged. The complete linkage clustering fits in KG canonicalization as demonstrated in (Vashishth et al., 2018b) as small-sized clusters are expected as opposed to single and average linkage clustering. Figure 6.6 illustrates the canonicalized version of Figure 6.5, where nodes are merged after the clustering process.

The final phase is determining a representative for each cluster. In line with the work of (Vashishth et al., 2018b) we calculate the mean of all the generated elements' embeddings weighted by the number of occurrences of each element in the input. The entity with the minimum distance to the weighted cluster mean is selected as a representative.

To further clarify the importance of canonicalization in addressing ambiguity and redundancy, consider the following two triple extractions: $\ll$ *Barack Obama, born in, Hawaii* $\gg$ and $\ll$ *Obama, served as,* 44th *U.S President* $\gg$. In an uncanonicalized version of the KG, the two extractions would be included separately without any connecting edges, as *Barack Obama* and *Obama* are perceived as two distinct entities. This may lead to a remarkable impact when querying data from the KG as it will not return all information linked with Barack Obama. Such KGs will also suffer from redundant facts, which are undesired. Canonicalizing KGs using HAC clustering as described above guaran-

tees relation transitivity, that both entities —*Barack Obama* and *Obama*— are fused to represent a single entity. Several other canonicalization approaches and entity linking techniques are proposed in (Hachey et al., 2013; Ceccarelli et al., 2013).

6.4 Results and Evaluation

In this section, we report the utility of Open-CyKG. KG curation is the task of assessing the value of the constructed KG, this process is commonly fulfilled by experts in the field. However, nowadays, it is deemed a tedious task to be done by humans, especially with densely populated KGs or even more complicated ones incorporating several domains (Fensel et al., 2020). Nonetheless, KG validation is still an open challenge, hence we address this matter by evaluating each component in our model separately to reflect the quality of Open-CyKG. We also present a set of auxiliary experiments to further analyze our proposed OIE and NER models. All our experiments were implemented using the Keras framework (Chollet, 2015) with the TensorFlow backend (Abadi et al., 2016). We start by describing the dataset used to build the KG in detail, along with its inherent constraints.

MalwareDB Dataset

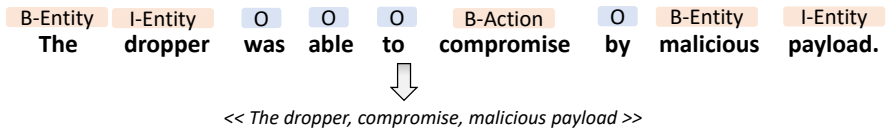

Figure 6.7: An example of OIE performed on a sentence from APT reports, where the Action tag represents the relation that links the two entities together.

As the world is digitally growing, devices are more prone to malware attacks which might lead to misfortunate events ranging from unauthorized access to personal data to device damage. MalwareDB (Lim et al., 2017) is an annotated dataset based around Malware Attribute Enumeration and Characterization (MAEC) vocabulary that primarily outlines malware characteristics gathered from 39 APT reports. Figure 6.7 shows a sample from the aforementioned dataset with the extracted triples from the OIE task.

OIE is the primary building block in Open-CyKG to construct the KG, so our main objective is to effectively identify relation triples necessary for successful querying. Hence, training data is crucial in our work. Unfortunately, one of the ongoing challenges is the lack of BIO-labeled data, specifically in understudied domains such as cybersecurity. Although the 39 APT reports that constitute the MalwareDB dataset originally contain 6,819 sentences, we were only able to classify 1,910 sentences as informative sentences, which challengingly formed our training, test, and validation sets. Uninformative sentences can be defined as:

- Sentences that are composed of 'O': Outside labels only.
- Phrases without any relationship labels.
- Sentences that contain only a single entity.

Nevertheless, currently, there is no alternative dataset available in the cybersecurity domain with BIO labeling.

Experimental Results and Analysis: OIE

In this section, we assess the outcome of our proposed attention-based OIE model, and we follow the framework configuration and dataset as discussed in Sections 6.3 and 6.4 respectively.

Word Embeddings

In recent years various types of embeddings have been proposed, varying from character and word embeddings to sentence and document embeddings. Nonetheless, they all provide the same function of mapping textual input to semantically meaningful vector representations. The innovative contextualized word embeddings are capable of capturing dense semantic and syntactic features of a word by incorporating context into the generated embeddings. In our OIE task, three inputs are embedded and fed to the network, making the choice of embedding fundamental. As a result of this, we opted to experiment using different word embedding techniques by selecting one conventional non-contextualized embedding —GloVe (Pennington et al., 2014)— and three contextualized embeddings with varying dimensionality and trained on a diverse set of domains. In order to carry out our experiments, we utilized Flair (Akbik et al., 2019), a robust open-source framework developed by Zalando Research that provides a unified interface for a wide range of state-of-the-art

Table 6.1: Four embedding techniques and their respective dimensionality employed in our OIE model.

Embedding Technique	**Vector Dimensionality**
GloVe (Pennington et al., 2014)	100
BERT (Devlin et al., 2019b)	3072
XLNet (Yang et al., 2019)	2048
XLM-RoBERTa (Conneau et al., 2020a)	1024

word, document, and sentence embeddings. The employed embeddings, along with their dimensionality, are shown in Table 6.1.

Experiment and Evaluation on MalwareDB

To assess the competence of our model, we have analyzed it rigorously with different experimental setups and word embeddings. As observed in Table 2, Bi-GRU achieves an overall higher score than Bi-LSTM neural network models. More precisely, our Bi-GRU + Attention model scores an F-measure of 59.4%, which is 2.2% higher than when using a Bi-LSTM + Attention network. Both achieved the highest score with XLM-RoBERTa embeddings.

We performed an ablation study to measure the effectiveness of the attention mechanism by testing on a Bi-GRU network which resembles the model introduced in Chapters 3 and 4. By removing the attention component, the Bi-GRU model achieved an F-measure score of 56.8%, verifying the impact of deploying the attention mechanism as it contributed to a 2.6% increase in F-measure. Despite the fact that GRUs and LSTMs capture long-range dependencies better than traditional RNNs, they do not have the ability to direct the focus to some of the input words to point out the words that are important to our task, which further demonstrates the importance of deploying attention mechanisms in information extraction tasks.

To further evaluate the potential of our attention-based OIE model, we compare our model against yet another prior state-of-the-art neural OIE network that is composed of Bi-LSTM as proposed by (Stanovsky et al., 2018). Similar to the comparison to the previous state-of-the-art, our model was able to achieve a higher F-measure by 4.2%. BERT embeddings attained the highest results in both networks, Bi-LSTM and Bi-GRU.

The rationale behind Bi-GRUs performing better than Bi-LSTMs is due to GRUs' ability to expose the complete memory, unlike LSTMs. Additionally, LSTMs have more gates than GRU, which causes the gradients to flow through,

which leads to steady progress being more complex to maintain after many epochs (Dey and Salem, 2017). Another interesting conclusion that can be drawn from our ablation study is the adoption of an attention mechanism to fully leverage the bidirectional context information, as it is also elaborated by the authors of (Bao et al., 2020) and (Vaswani et al., 2017).

It should be emphasized that despite the fact that achieving a decent recall and precision would be the optimal situation, precision is more crucial in our work as it reflects the certainty of the extracted information. In KG, false positives are expensive to maintain, in a setting of a high-scoring recall and a lower precision, the KG would be populated with uninformative or false information, which will result in a less-efficient querying experience. Nevertheless, it is important to note that the highest precision achieved by our model was 62.9% using BERT, trading off to a lower recall of 54.7%, which resulted in a decrease of 0.9% in F-measure when compared to our highest achieving score of 59.4% reported in Table 6.2.

Our model's results are sensitive to hyperparameter alternations, thus, a grid search was performed to single out the optimal number of training epochs and batch size. The hyperparameter configurations that realized the best results are reported in Table 6.3. The hidden size of all Bi-GRU layers is set to 128, which is also the same number of units used in the two TDD layers. For regularization reasons to prevent over-fitting, the dropout rate is adjusted to 0.1. Moreover, early stopping is employed to terminate training based on the model's performance on the development set. Furthermore, a linear activation function, Rectified Linear Unit (ReLU) (Nair and Hinton, 2010b) was applied in the two TDD layers, while the Adam optimizer (Kingma and Ba, 2015) was utilized for training our model.

Additionally, as the limited size of the training set influences the neural network's performance, we were able to further evaluate our proposed attention-based OIE model by experimenting on a larger annotated news dataset (Stanovsky and Dagan, 2016), which is composed of 2906 training sentences. An increase of 3.2% in F-measure was achieved, emphasizing that the limited size of training data indeed plays an important role.

Experimental Results and Analysis: NER

As the MalwareDB dataset has no named entities annotation, we could not train or evaluate our model based on MalwareDB extractions. In this section, we will discuss the datasets used to train and validate our NER neural network described in Section 6.3.

Table 6.2: Results of our Open-CyKG: OIE model (Open-CyKG$_{\text{Bi-GRU+Att}}$). Recall, precision, and F-measure are used as evaluation metrics. Along with the deployed word embedding that resulted in the highest scores.

Model	**Network Architecture**	**Word Embedding**	**Results**		
			Recall	Precision	F-Measure
(Sarhan and Spruit, 2019), (Sarhan and Spruit, 2020)	Bi-GRU	BERT	54.9 %	58.9 %	56.8 %
(Stanovsky et al., 2018)	Bi-LSTM	BERT	53.0 %	57.5 %	55.2 %
Open-CyKG$_{\text{Bi-LSTM+Att}}$	Bi-LSTM + Attention	XLM-RoBERTa	55.7 %	58.7 %	57.2 %
Open-CyKG$_{\text{Bi-GRU+Att}}$	**Bi-GRU + Attention**	**XLM-RoBERTa**	**57.2 %**	**61.8 %**	**59.4 %**

Table 6.3: Hyperparameter settings used in our OIE model.

Hyperparameter	Value
Epochs	100
Batches	50
Bi-GRU	128 units
TDD Activation Function	ReLU
TDD units	128 units
Dropout Rate	0.1
Optimizer	Adam

Dataset

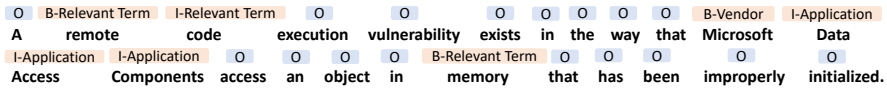

Figure 6.8: Sample from the Microsoft Security Bulletin dataset with the named entity tags following the BIO tagging schema.

To perform NER, which is necessary for refining and labeling KG nodes, we used two different training sets, each containing a diverse set of labels. The output from the two-pass process is then merged to assign labels from the two datasets to named entities. The first dataset Microsoft Security Bulletins, utilized in (Bridges et al., 2013), is also annotated and made widely accessible by the authors. It is composed of 5072 sentences discussing vulnerabilities and security flaws in Microsoft's software products along with various patch and mitigation information. A random sample drawn from the Microsoft Security Bulletin dataset is shown in Figure 6.8. The second dataset employed in (Kim et al., 2020) is a malware-specific dataset collected from various CTI reports resulting in a total of 3450 sentences with predefined named entities that relate to malware, which are considered key elements in our malware-based APT reports. The predefined tags in both datasets with their respective ratio in the training set are illustrated in Figure 6.9. It is worth noting that when two labels are assigned to a single word, we select the malware-specific CTI report label since it is more closely related to our APT reports dataset.

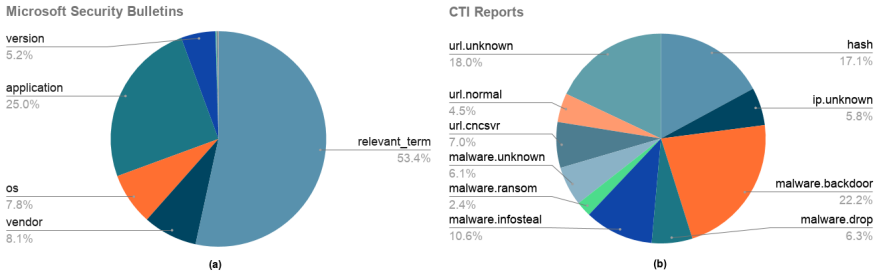

Figure 6.9: (a) Labels distribution of the Microsoft Security Bulletin training dataset (Bridges et al., 2013). (b) Labels distribution of the CTI training dataset (Kim et al., 2020). Note that label 'O': Outside, takes up the majority of the words' labels but it's removed from the chart for better illustration of the distribution of informative tags.

Experiments and Evaluation

For both datasets, we split the corpus into two partitions, 80% for training and 20% for testing, with setting apart a fraction of 0.1 of the training set for validation purposes to assess the loss at the end of each epoch. To certify the quality of our model, we performed five-fold cross-validation. We opted for stratified K-fold as it takes the cross-validation process one step further by preserving the distribution of the class in both training and test splits to avoid unbalanced labels' distribution. To validate the efficiency of our NER network architecture, we compared our model's performance against the results of different baselines and state-of-the-art models.

Table 6.4 shows the complete results of our NER model that is employed in Open-CyKG on the Microsoft Security Bulletins dataset. We can clearly see that the proposed model outperforms both the baselines and state-of-the-art models by scoring a 98.5% F-measure. More precisely, when comparing our Bi-GRU + CRF results to the baseline reported by (Bridges et al., 2013) that employed a traditional approach using hand-crafted rules, we can see that our model outperforms by more than 20% of the F-measure score. In addition, there is an increase of 1.9% in the F-measure when we compare our proposed Bi-GRU model to the Bi-LSTM network architecture with 50 batches and 30 epochs.

It is observed that all models obtain decent precision, however, the overall performance of NN models significantly outperforms hand-crafted rules. The low recall achieved by (Bridges et al., 2013) model contributed to this decrease.

Table 6.4: Results of the Open-CyKG: NER model (Open-CyKG$_{\text{Bi-GRU}}$). Both training and testing are done on the Microsoft Security Bulletin dataset. Along with the original dataset baseline results as reported in (Bridges et al., 2013) and Bi-LSTM + CRF network. Recall, precision, and F-measure are used as evaluation metrics.

Model	**Method**	**Results**		
		Recall	Precision	F-Measure
Bridges et al. (2013)	Hand-crafted Heuristic	75.3 %	**99.4 %**	77.8 %
Open-CyKG$_{\text{Bi-LSTM}}$	Bi-LSTM + CRF	96.6 %	97.4 %	97.0 %
Open-CyKG$_{\text{Bi-GRU}}$	**Bi-GRU + CRF**	**98.7 %**	99.2 %	**98.9 %**

The rationale behind this is due to the considerable variation of expressions in the natural language specifically reflected in tasks such as NER an RE, as it is also shown in the work of (Nadeau and Sekine, 2007).

The performance of our NER model on the second dataset —— CTI reports —— is presented in Table 6.5. Three comparisons are carried out in this experiment; the first is the baseline model that originally annotated and constructed the CTI reports dataset, which employed a BOC-based Bi-LSTM + CRF network architecture proposed by (Kim et al., 2020). The second is a character-based CNN that also uses a Bi-LSTM architecture. A pure Bi-LSTM + CRF network is our third comparison that is trained with the same hyperparameters of our model. As it is observed when comparing our model with the baseline, our model scored an F-measure of 79.8%, which is 4.7% higher than that of (Kim et al., 2020), the increase was mainly reflected by an increase in the recall of our model by 10.3%. In addition to reporting the findings of their model on the CTI reports dataset, the authors of (Kim et al., 2020) reported the score of using a CNN-based Bi-LSTM + CRF network —— the second comparison —— which resulted in almost the same score as the BOC system and 4.8% decrease in F-measure when compared against the score of our model. Furthermore, in line with the evaluation we carried out on the Microsoft Security Bulletin dataset, we implemented a state-of-the-art Bi-LSTM+CRF network which achieved the lowest score among all other models reported in Table 6.5 by scoring an F-measure of 71.1% on 10 training epochs with a batch size of 50.

In addition to the reasons mentioned in Section 6.4 on why Bi-GRUs outperform Bi-LSTMs, in this experiment, we attribute the increase to the nature of the CTI dataset used in this NER task, which is a small dataset with long sentences. This phenomenon is also observed in the work of (Yang et al.,

Table 6.5: Results of our Open-CyKG:NER model (Open-CyKG$_{\text{Bi-GRU}}$). Both training and testing are done on the CTI dataset, and the original baseline results in (Kim et al., 2020) are reported along with the CNN-based network. Recall, precision, and F-measure are used as evaluation metrics.

Model	**Network Architecture**	**Results**		
		Recall	Precision	F-Measure
Kim et al. (2020)$_{\text{BOC}}$	BOC: Bi-LSTM + CRF	70.5 %	**80.3 %**	75.1 %
Kim et al. (2020)$_{\text{CNN}}$	CNN + Bi-LSTM + CRF	71.0 %	78.9 %	75.0 %
Open-CyKG$_{\text{Bi-LSTM}}$	Bi-LSTM + CRF	70.4 %	71.9 %	71.1 %
Open-CyKG$_{\text{Bi-GRU}}$	**Bi-GRU + CRF**	**80.8 %**	78.9 %	**79.8 %**

Table 6.6: Hyperparameter settings used in our NER model on both datasets.

Hyperparameter	**Microsoft Security Bulletins**	**CTI Reports**
Epochs	30	10
Batches	50	30
Bi-GRU	50 units	50 units
TDD Activation Function	ReLU	ReLU
TDD units	50 units	50 units
Dropout Rate	0.1	0.1
Embedding	Keras	Keras
Optimizer	Adam	Adam

2020a).

Table 6.6 states our hyperparameter configurations that realized the reported scores in Tables 6.4 and 6.5 for both datasets. Bi-GRU layers and TDD layers have a mutual number of units —50— with ReLU selected as an activation function in the TDD layer. Similar to our OIE network, the dropout rate is set to 0.1 to prevent over-fitting, and early stopping is utilized. All models are trained with the Adam optimization algorithm.

Although the MalwareDB dataset has no annotated named entity tags, the results indicate that our model can effectively label cybersecurity-related terms. To further validate the performance of our NER model, a random sample of 10% was selected from the MalwareDB test set and manually annotated to compare against the model's predicted labels. When training on the Microsoft Security Bulletins dataset, our model achieved a recall, precision, and F-measure of 85.5%, 87.7%, and 86.6%, respectively. In comparison, training

on the CTI reports resulted in a recall, precision, and F-measure of 83.6%, 82.3%, and 82.9%, respectively.

Canonicalization Evaluation

To evaluate canonicalization using contextualized word embeddings carried out in Open-CyKG, we manually construct a gold standard of clusters that represents the ground truth clusters of all extracted entities. We follow the work of (Vashishth et al., 2018b; Wu et al., 2020; Galárraga et al., 2014) by using *macro*, *micro* and *pairwise* metrics to evaluate canonicalization. We concisely explain these metrics below. Let C be the clusters produced by Open-CyKG canonicalization, and G denotes the gold standard clusters.

Macro: Macro precision (P_{macro}) can be defined as a fraction of pure clusters in C formed by our approach that are linked to the same gold standard G. While Macro recall (R_{macro}) is the inverse of (P_{macro}), by interchanging the roles of C and G as seen in Equations 6.3 and 6.4.

$$P_{macro}(C, G) = \frac{|\{c \in C : \exists g \in G : g \supseteq c\}|}{|C|} \tag{6.3}$$

$$R_{macro}(C, G) = P_{macro}(G, C) \tag{6.4}$$

Micro: Micro precision (P_{micro}) measures the purity of the clusters C under the assumption that the most frequent gold entity of the mentions in a cluster is the correct entity (Schütze et al., 2008), as depicted in Equation 6.5, where N denotes the number of mentions in the input. In a similar manner, micro recall (R_{micro}) is the inverse of (P_{micro}) as shown in Equation 6.6.

$$P_{micro}(C, G) = \frac{1}{N} \sum_{c \in C} max_{g \in G} |c \cap g| \tag{6.5}$$

$$R_{micro}(C, G) = P_{micro}(G, C) \tag{6.6}$$

Pairwise: A *hit* in cluster C indicates that two mentions refer to the same gold entity. Pairwise precision ($P_{pairwise}$) measures the ratio of the number of hits ($\#hits_c$) in C to total possible pairs ($\#pairs_c$) in C (Vashishth et al., 2018b), where $\#pairs_c = |c| * (|c| - 1)/2$. Equations 6.7 and 6.8 define pairwise precision ($P_{pairwise}$) and pairwise recall ($R_{pairwise}$) respectively.

$$P_{pairwise}(C, G) = \frac{\sum_{c \in C} \#hits_c}{\sum_{c \in C} \#pairs_c} \tag{6.7}$$

$$R_{pairwise}(C, G) = \frac{\sum_{c \in C} \#hits_c}{\sum_{g \in G} \#pairs_g} \tag{6.8}$$

In all cases, F-measure is defined as the harmonic mean of the model's precision and recall. The optimal threshold value chosen for HAC clustering was decided upon using a grid search on the validation set. Due to the fact that XLM-RoBERTa was the highest-scoring language model in our OIE phase, we used the generated embeddings to compute the distance metric. It should be emphasized that the word embeddings used in the clustering phase are generated based on the whole input sentence to fully leverage the concept of contextual embeddings. Results are shown in Table 6.7. We observe that the recall achieved in all metrics is moderate to good, in line with canonicalization results reported in related works (Vashishth et al., 2018b; Wu et al., 2020; Galárraga et al., 2014). Nonetheless, If we look at pairwise precision, we can notice it is relatively low, which indicates that not all pairs of entities in C refer to the same gold entity.

Table 6.7: Canonicalization results.

Metric	**Results**		
	Recall	Precision	F-Measure
Macro	86.5 %	78.9 %	82.6 %
Micro	90.5 %	74.4 %	81.7 %
Pairwise	79.6 %	54.7 %	64.8 %

Demonstrating Information Retrieval using Open-CyKG

In this section, we present two sample queries, a general one targeting malware, while the other is a more specific query that focuses on watering hole attacks. To further illustrate the effect of the applied fusion technique, we perform an ablation analysis by solely applying the first phase of our refinement process as explained in Section 6.3. The ablation analysis is supported by analyzing the results of the queries to inspect the outcome of word embeddings in Open-CyKG canonicalization.

Cypher (Francis et al., 2018) is the official supported query language in Neo4j. It is an SQL-inspired declarative querying language that permits users to retrieve data from a graph database. The queries performed along with their Cypher translations are shown in Figure 6.10.

Query 1:
What are the properties of malware attacks ?

Cypher:
```
MATCH (n1:E1 {Name: 'malware'})
MATCH (n2:E2)-[r:RELATION]->(n1)
RETURN n1.Name, r.Name, n2.Name
```

Information retrieved from non-canonicalized KG:
- Malware identify and unmask the IP address of target(s)
- Malware uses the old PKWARE implementation of zip encryption.
- Malware use location awareness
- Malware via e-mail *
- Malware sends data captured by the malware *
- Malware introduced to destination network
- Malware autonomously infect removable drives, like USB sticks, or project files for PLCs

Additional results retrieved from canonicalized KG:
- Malware profiled the victim's computer
- Malware beaconed it's IP-address
- Malware infects a user who views the decoy
- Malware serves as a beacon
- Malware sends the data in smaller bursts

Query 2:
How can attackers use watering hole attacks ?

Cypher:
```
MATCH (n1:E1)-[:CONTAINS]-()-[c2:CONTAINS]-(e3)
MATCH (n2:E2)-[r:CONTAINS]->(n1)
WHERE n1.Name='attackers' AND n2.Name='watering hole attacks'
MATCH (n2)-[y:CONTAINS]->(c2)
RETURN e1.Name, r.Name , n2.Name, c2.Name, e3.Name
```

Information retrieved from non-canonicalized KG:
- Attackers use watering hole attacks to infect their victims *

Additional results retrieved from canonicalized KG:
- Attackers run a vast network of watering hole attacks to target visitors with surgical precision

Figure 6.10: Two sample query results as retrieved from Open-CyKG.

In the first query, Open-CyKG was able to retrieve diverse information as long as it was directly connected to a 'malware' node. The majority of the retrieved data capture valuable insights on malware threats, whereas in some cases the extraction can be considered less informative such as *'Malware via e-mail'* , or uninformative such as 'Malware sends data captured by the malware'. In the second query, the retrieved data had to relate to both 'attackers' and 'watering hole attacks', since this query has a higher level of specificity, it only results in two extractions, although *'Attackers use watering hole attacks to infect their victims.'* might be interpreted as an uninformative extraction. Nevertheless, when the queries are performed on the canonicalized version of Open-CyKG, the KG is able to deliver more insights on the requested data. Additionally, as the generated named entity tags discussed in Section 6.4 are assigned as properties to nodes and edges, they can be leveraged to eliminate uninformative or ambiguous extractions while querying.

As observed, canonicalization using contextualized word embeddings aids in capturing more information. As a result, security analysts can query Open-CyKG to retrieve data on a specific cyber entity, albeit being expressed differently among various APT reports.

6.5 Conclusion

We introduced Open-CyKG: a novel framework that combines features from several components along with fusion techniques using contextualized embeddings to generate a knowledge graph from advanced persistent threat reports. Our proposed framework is developed from two core components, an attention-based OIE and a cybersecurity NER system.

We validated the quality of the generated KG by evaluating each component separately. We evaluated our cybersecurity NER model against several baselines and state-of-the-art models on two different datasets. Our model was able to deliver the best performance. The attention mechanism is a revolutionary theory that transformed the way researchers design neural networks. Not only does it have an essential role in various neural network-based NLP tasks to enhance performance, but it also offers important insights into how the models are operating. This has motivated the development of our attention-based OIE framework, which we validated by performing an ablation study and compared against state-of-the-art OIE models. In both cases, our attention-based model achieved the best results. Another interesting option would be a transformer-based model. A transformer-based model relies on self-attention, at each time step, there is direct access to all other steps, which practically means that there is no room for information loss. However, most transformer-based models have a quadratic complexity, which limits the token length as a trade-off between performance and memory usage, resulting in truncating training sentences (Fan et al., 2020). The authors of (Han and Wang, 2021) introduced a transformer-based OIE model, however, its performance was not evaluated against any state-of-the-art neural network model. Thus, as a future direction, we intend to further evaluate different neural OIE research trends, including transformer-based models on benchmark datasets.

Despite their value and practicality, KGs usually suffer from incompleteness, redundancy, and ambiguity that might translate to uninformative query results. First and foremost, we are in the process of acquiring more cybersecurity data from AAR reports to carry out a large-scale experiment. In future work, we will shift our attention to KG completion and link prediction to further enhance the strength of the generated KG. Additionally, we would like to explore the possibility of extending the KG model to include a dynamic reasoning component instead of entirely relying on static information to build the KG. A final exciting addition would be constructing a multi-lingual or cross-lingual KG to support machine translation-based applications. All in all, we foresee that in the near future KGs will become sufficiently mature to

provide added value to daily practices in cybersecurity and beyond.

7 | Conclusion

Motivated by the potential of Open Information Extraction (OIE) in downstream Natural Language Processing (NLP) tasks, the goal of this dissertation has been the study of the deployment of the OIE task in knowledge representation to model intelligent behavior that is beneficial for data-intensive applications.

This work takes as a premise that the design choice for the OIE task heavily contributes to the value of the extracted information, as was proven by the presented quantitative analyses in the previous chapters. We have shown how OIE can contribute to three areas: (i) extraction of textual data from unstructured text, (ii) transfer learning across different domains and tasks, and (iii) Knowledge Graph (KG) population.

Additionally, we have demonstrated that our proposed methodology for KG construction — *Open-CyKG in Chapter 6* — is able to extract vital information without requiring a predetermined set of relations or utilizing an existing ontology. As a result, this decreases the probability of missing out on crucial knowledge as the extraction is not limited to a specific set of information. We further demonstrate that the design decision of the Open-CyKG components significantly impacts the overall pipeline and contributes to an effective and efficient knowledge representation. This has been one of the primary aims of this dissertation, to develop a deep understanding of the design choices of the components that constitute the knowledge representation pipeline, with a primary focus on the OIE component, and thus, expectantly, enable users to use it in a way that works best for them. At the beginning of this research, we posed the following main research question:

MRQ — To what extent can we enhance open information extraction methods to efficiently and effectively represent unstructured textual data?

In order to answer this question, we first elaborated on several IE techniques in Chapter 1, along with explaining the need to have an efficient knowledge representation. Then in Chapter 2, we surveyed state-of-the-art approaches in OIE, and we thoroughly overviewed their key design choices, along with the gains and shortcomings of each approach. We found that conventional

OIE methods rely either on machine learning-based classifiers or hand-crafted rules. Each method is further sub-categorized into approaches that employ shallow syntactic analysis or dependency parsing techniques. Moreover, we discussed the challenges associated with OIE along with future directions.

In Chapters 3 and 4 we investigated, developed, and evaluated several Recurrent Neural Network (RNN) OIE models. These two chapters demonstrated the strength of employing a Bidirectional Gated Recurrent Unit (Bi-GRU) OIE network to extract relation triples. Precisely, we investigated the effect of contextual word embeddings on OIE in Chapter 3. Chapter 4 subsequently built on the Bi-GRU OIE model as we conducted several experiments toward model generalization. In more detail, we investigated the transferability of the features learned from training a neural network on an OIE task to both a different domain and a semantically related task. An important point we came across while conducting our research is that the presence of long-term dependencies might affect the model's adaptability.

Chapter 5 is primarily built on the transfer learning foundation. There is a growth in the number of studies to investigate the cognitive plausibility of computational linguistics to explore the ability of modern language models to identify different linguistic structures at the semantic or syntactic level. In this chapter, we explored pre-trained language models for taxonomy classification and proposed *UU-Tax*. We conducted a deep analysis in multi-task learning, data-enriched fine-tuning, and multi-stage fine-tuning for the classification sub-task to overcome the instability of pre-trained language models and improve their robustness. However, these approaches were proven to be ineffective in the taxonomy regression sub-task; alternatively, we explored the use of several machine learning regressors.

Finally, Chapter 6 assembles the research outcome of the previous chapters to introduce *Open-CyKG*, an attention-based OIE model for knowledge graph construction. As a case study, we utilized unstructured cyber threat intelligence reports. With the support of knowledge representation, particularly knowledge graphs, we can discover correlations across different objects from different big data sources, construct semantic connections between these objects, and effectively turn big data into usable knowledge. Knowledge graphs can help us improve the quality, productivity, and adaptability of the decision-making process in knowledge-intensive applications as it does a good job at relating information together as more data goes in.

7.1 Contributions

In Chapter 1, Introduction, we posed five research questions to investigate various aspects of the main research question. In this section, we review the contributions of each chapter to answer these research questions and formulate a conclusion for each one. Together, they contribute to answering this dissertation's main research question.

RQ1 — What are the existing OIE methods and challenges faced to advance triple extraction and information retrieval from unstructured texts?

An immense amount of textual data is still organized in an unstructured format, which obstructs the usability of this data in any downstream application. Fast and scalable IE techniques to detect, extract and organize vital information are needed to make the most out of these massive amounts of data. Nonetheless, several challenges are associated with IE techniques, such as their limitation when implemented on a large scale across several domains. This is when OIE swings into action. By nature, OIE is an unsupervised IE technique as it is an open-domain and relation-independent paradigm that results in scalable and fast behavior. OIE extracts factual information from the text in the form of $\ll$ *argument 1, relation, argument 2* $\gg$ without the need for a pre-defined relation set. It is worth noting that the extracted tuples are not only limited to binary extractions but can also be extended to n-ary relations.

Using the snowballing method, we provided a detailed overview of conventional OIE models that are either machine learning-based or rule-based approaches. We further subdivided each approach into two different categories, either approaches heavily dependent on shallow syntactic analysis or approaches that rely on dependency parsing. We started by reviewing the first implemented OIE system, TextRunner, a self-supervised learning approach introduced in 2007 (Yates et al., 2007). With the exclusion of TextRunner, the research was limited to articles that were published in the period between 2008-2018. Back then, deep learning approaches in OIE were on the verge of development. Additionally, recent developments in neural OIE models were covered in Chapter 6.

After extensively reviewing traditional OIE approaches, we concluded that the self-supervised nature of the machine learning-based approaches mitigates the need for defining a new set of rules when shifting across domains. Nevertheless, learning-based approaches are more error-prone when compared to

rule-based approaches as they result in uninformative and incoherent extractions. Recently, neural-based approaches transformed the OIE task in which the problem is mainly formulated as a sequence labeling task and lessened the challenges faced by the aforementioned methods.

Findings I — Prior to the deep learning and neural networks era, conventional OIE systems were either learning-based or rule-based approaches. Supervised machine learning methods necessitate labeled data for training, which can be time-consuming and expensive to annotate. However, once trained, these models are able to extract relations in a domain-agnostic and scalable manner, reducing the need for human involvement. Rule-based OIE techniques, on the other hand, tend to produce more accurate, informative, and coherent extractions but require significant manual effort. We demonstrate the evolution of OIE approaches over time and the specific challenges they address. Our findings are consistent with the work of Niklaus et al. (2018b), who presented an overview of the various methods proposed to solve the task of OIE and classified them into the four categories of learning-based, rule-based, clause-based systems, and approaches capturing inter-proposition relationships, thereby showing their evolution over time as well as the specific problems they tackle. Despite the trade-offs of these earlier approaches, deep neural networks are emerging as promising approaches for OIE systems.

RQ2 — How can contextualized word embeddings enhance the process of Open Information Extraction?

OIE lacks a thorough analysis of how different deep learning models perform and the effect of contextualized word embeddings in the task of OIE. In this regard, we developed and investigated diverse neural network models to better understand the contribution of deep contextual embeddings in improving the extractions relative to classical pre-trained embeddings.

We evaluated the results of three different networks, Bidirectional Long Short-Term Memory (Bi-LSTM), Bidirectional Gated Recurrent Unit (Bi-GRU), and Hierarchical Attention Network (HAN), whilst employing static and contextual embeddings, namely Global Vector word representations (GloVe) and Embeddings from Language Models (ELMo), respectively. The experimental design we used included a grid search optimizing hyperparameters. We opted to use a Bi-GRU in our proposed OIE model as it has two main advantages; it is less complex compared to LSTM as GRU controls the flow

of data in a similar way as LSTM but without having to utilize a memory unit, thus making it more efficient, which ultimately leads to faster training. Further, for long-range dependencies and relations, GRUs have proven their effectiveness, as is also elaborated in (Kaiser and Sutskever, 2015; Yin et al., 2017). To train our model, we used two datasets, Newswire and WikiNews, containing 1241 and 1957 sentences, respectively.

As challenging as it was to train a model with a relatively small dataset, we conducted extensive experiments to compare different model variants and baselines. We were able to demonstrate that our proposed Bi-GRU model effectively extracts relation triples. Moreover, ELMo has contributed significantly to the recent developments of NLP. ELMo is clearly at the forefront of various NLP tasks following up on the findings of (Young et al., 2018).

Findings II — A Bi-GRU-based OIE model with contextual word embeddings, in which the task is formulated as a sequence tagging problem, offers better performance than existing state-of-the-art algorithms. This is in line with previous research by Stanovsky et al. (2018), who also demonstrated the effectiveness of formulation of OIE task as a sequence tagging task. The evaluation measures — precision, recall, and F1 score — were used for quantitative evaluations to provide a thorough assessment of the proposed OIE model. This demonstrates that context-dependent embedding models, in particular ELMo, are superior to other textual representation techniques. Given the obtained quantitative results, we recommend GRU models in information extraction systems as a promising research direction, given their ability to capture long-range dependencies and store and filter information. Overall, our contributions highlight the importance of using state-of-the-art deep learning techniques and contextualized embeddings for improving the performance of OIE models, and provide further evidence for the benefits of Bi-GRU models and ELMo embeddings in particular.

RQ3 — How transferable is OIE to other NLP tasks to diminish the complication of insufficient training data of neural network models and aid in model generalization?

Whereas Chapter 3 established that deeply contextualized embeddings are a promising approach to representing text representation in the OIE task, Chapter 4 starts by further investigating the deployment of other contextual word embeddings to determine which model better suits our data and task.

The premise of conventional Machine Learning (ML) techniques is that the training and the test datasets are both of the same domain, which entails that both the input feature space and data distribution properties have a high level of similarity. Nonetheless, this assumption does not hold in several machine learning scenarios as the process of collecting a dataset that is suitable for training an ML model. Therefore, it is crucial to develop high-performance learners that are domain agnostic.

The experiments we conducted to answer this research question covered two aspects; transferring to a new domain and transferring to a semantically related task. Each experiment was further subdivided into two modes, transductive learning, where we have labeled data in the source task only, and inductive learning, in which we have labeled samples from both the source and target tasks. We started our research by transferring the features from the OIE news domain to a bio-medical domain. The result obtained from inductive transfer learning was only 1% lower than conventional learning in the F-measure score. While experiments on related end tasks such as RE resulted in a difference of only 3.8% of F-measure score when relying on inductive transfer learning instead of traditional learning. When relying entirely on the OIE news dataset to train our model in a transductive setting, we achieved a lower F-measure score by 2% and 2.5% on the bio-medical domain and RE task, respectively, when compared against the inductive setting. Our results demonstrate that OIE can be utilized as a reliable source task due to similarity in features with the RE task, as the model is able control which information is taken into account. This contributes to overcoming the lack of labeled datasets for training neural models.

Findings III — To evaluate the transferability of the OIE task using the proposed Bi-GRU model, we perform a set of experiments to study how can OIE be utilized in low-resource scenarios. Results show that our OIE model using transfer learning has the ability to adapt to a new domain. Another promising observation is that our findings demonstrate that OIE can significantly benefit the research across interrelated NLP tasks. Our findings suggest that OIE can serve as a powerful pre-training step for related NLP tasks, ultimately leading to more accurate and efficient models, which is deemed as a starting point for future research to examine transfer learning of OIE beyond semantically related tasks, as our work offers a valuable contribution to the understanding of transfer learning in the context of OIE.

RQ4 — How can we enhance the robustness of pre-trained language models to generalize for unseen patterns during inference on downstream tasks?

As taxonomies are becoming essential to a large number of NLP applications and knowledge management, it has been receiving an increasing amount of attention in recent years. However, the majority of the state-of-the-art approaches wrongfully assume that the taxonomic relationship that holds between two nouns in a sentence is correct, which leads to undesired error propagation in downstream applications. Chapter 5 details our participation — as UU-Tax — in the SemEval-2022 task PreTENS (Presupposed Taxonomies: Evaluating Neural Network Semantics). PreTENS tackles one of the ongoing challenges in taxonomy end tasks, which is taxonomy prediction. PreTENS is articulated into the two following sub-tasks: A binary classification sub-task, which involves predicting the acceptability label assigned to each sentence of the test set, and a regression sub-task, which entails predicting the average score assigned by human annotators on a seven-point Likert scale with respect to the subset of data evaluated via crowdsourcing. The task covered three languages: English, French, and Italian. Fine-tuning a pre-trained model requires an adequate amount of labeled data to fulfill its potential on downstream tasks. The significant challenge of this task is the scarcity of annotated data, which is correlated with the diversity of the information enclosed within this data. To elaborate further, only 28.6% of the dataset available for sub-task 1 was accessible for training. As pre-trained language models fail to generalize on unseen patterns during inference, we formulated this task to answer research question #4, and thus explored several approaches to improve the robustness of the pre-trained language models.

Through a series of experiments we investigated the performance of several models using three state-of-the-art transformer approaches:

1. Multi-task fine-tuning using a correlated task, namely common sense validation.

2. Data-enriched fine-tuning using prompt-based learning by feeding extra features to the pre-trained language model during the fine-tuning phase.

3. Multi-stage fine-tuning by exploiting data augmentation techniques, specifically back-translation, insertion, and substitution techniques, to address annotation scarcity.

In addition to the aforementioned experiments, we performed a thorough ablation analysis to investigate further the effect of different data augmentation techniques in multi-stage fine-tuning. Experimental work has shown

that employing a two-stage fine-tuning model using the ELECTRA transformer yielded the best result in the first sub-task by achieving an F1 score of 92.84%. In the first stage, we fine-tuned ELECTRA using substitution and insertion techniques provided by the NLPAug tool. In contrast, in the second stage, fine-tuning was performed using back-translated data and the provided dataset.

Nonetheless, due to the nature of regression, we could not deploy the same methodology of the first sub-task and rely on insertion and substitution techniques for data augmentation to address the second sub-task. As we have elaborated in Chapter 5, pre-trained transformers did not perform well since the provided training set only contains as few as four distinct patterns. Thus, we proposed another model to address the regression sub-task that relies on features obtained from the Universal Sentence Encoder (USE) embeddings that are fed to regressor models. Our experiments covered different regressors; Logistic Regression (LR), K-nearest Regressor (KNR), Decision Tree (DT), an7d Support Vector Regressor (SVR). Experimental results showed that there is no one-size-fits-all regressor for all three languages due to lexical and semantic differences.

Findings IV — By applying data augmentation for taxonomy classification, we were able to provide valuable guidelines on the usefulness of multi-stage fine-tuning in pre-trained language models. However, the methodology deployed in the first subtask proved not adaptable to the regression sub-task. This shows that the deployment of pre-trained language models is still in its infancy regarding regression tasks with limited training data. Instead, a simple model that utilized features from a sentence encoder to train different regressors proved its efficiency in the regression sub-task. From a knowledge representation perspective, the SemEval-2022 shared task, PreTENS, demonstrates the importance of taxonomy classification prior to the construction of a knowledge base. Our work offers a valuable contribution to the understanding of the use of pre-trained language models in the context of taxonomy classification and regression tasks and suggests promising avenues for future research in this area.

RQ5 — How can OIE be employed to populate knowledge graphs to facilitate querying and retrieval of data?

In this digital era, there is a massive increase in textual information in any domain. This growth hinders the practicality and serviceability of existing

knowledge graphs, which are deemed incomplete. The majority of the already existing knowledge graph completion approaches are founded on the closed-world hypothesis, in which the knowledge graph is static. The reason behind this is the costs accompanying adding new entities and relations to update the knowledge graph. Maintaining an open-world knowledge graph would alleviate this problem; however, sustaining an open knowledge graph is still challenging as the textual data is presented in an unstructured format that may contain noisy or uninformative data. Open Information Extraction (OIE) is a pivotal component in the process of building an open knowledge graph as it does not require any predefined relations. The extracted triples from the OIE model are used to populate the knowledge graph. However, as those extractions might be uninformative, fusion and canonicalization techniques are required to maintain a reliable and efficient knowledge graph. This final question investigates how OIE can be utilized to construct a queryable knowledge graph. Our motivation to investigate cybersecurity as a case study is two-fold. First, several Cyber Threat Intelligence (CTI) reports provide essential information and insights to cybersecurity analysts. Nonetheless, limited research has been done on deploying NLP techniques in the cybersecurity domain compared to other fields. Second, taking advantage of cybersecurity-related information from various sources such as blogs and news websites necessitates an IE tool to extract vital information promptly; this perfectly fits our model proposal and research aims. To this end, we proposed a methodology that consists of three main phases:

1. An attention-based OIE system that extracts relation triples from unstructured malware reports.

2. A Bi-GRU-CRF Named Entity Recognition (NER) model that is trained on two distinct cybersecurity datasets to label prominent words in the cybersecurity domain.

3. Fusion and canonicalization techniques that rely on Hierarchical Agglomerative Clustering (HAC).

Validation of knowledge graphs is a crucial task to ensure the quality of information. It verifies the knowledge it holds as it measures the degree of accuracy of the enclosed data and ensures it corresponds with the so-called "real" world. However, manual validation has significant practical implications regarding large-sized knowledge graphs, making it a laborious and error-prone task. In addition, validation of the knowledge will not reflect the coverage of the information enclosed in the unstructured reports that were used to

populate the knowledge graph. On the grounds of this, we opted to validate the knowledge graph by validating each of the previously-mentioned phases separately.

Through a series of thorough experiments and ablation analyses, our proposed attention-based OIE mechanism was able to outperform the state-of-the-art models. Although our proposed Bi-GRU-CRF NER model resembles prior efforts, empirical experiments show that our presented model can achieve competitive performance with respect to both the baselines and state-of-the-art models. We had to manually create the gold standard clusters to evaluate the proposed fusion method that relies on HAC using features obtained from word embeddings. The evaluation performance metrics used in our experiments show that the proposed method can effectively address the knowledge graph fusion and canonicalization challenges.

Findings V — An end-to-end model for constructing a queryable Cyber Threat Intelligence (CTI) Knowledge Graph (KG) by primarily relying on OIE to extract relation triples from unstructured reports using the assistance of Named Entity Recognition (NER), shows that OIE has high potential to curate and construct KGs. We demonstrate that Open-CyKG contributes to building a KG in a fully-automated approach that does not require the need for any human involvement. We provided valuable and novel insights about the impact of HAC using contextual embeddings in KG canonicalization, as it is able to capture the semantics of noun phrases. This demonstrates the competence of embeddings-based methods in knowledge fusion in redundancy elimination. By demonstrating the potential of our proposed methodology through sample queries, our work offers a valuable contribution to the field of cybersecurity and the broader application of OIE in knowledge representation. Extensive experimentations demonstrated OIE's ability to develop an efficient and effective knowledge representation.

7.2 Insights for the NLP Community

Throughout our research, we have gained valuable insights that we believe can benefit the wider NLP research community. To make our contributions more accessible, we have brought together the elements of the previous sections in this section. Here, we highlight some of the key lessons we learned.

Evolution of OIE — Our work presented in Chapter 2 provides insights into the advancements of OIE methodologies and the specific challenges that

different methods address, as well as identifying existing gaps. Our review underscores the trade-offs associated with machine learning-based approaches and rule-based approaches, offering guidance to researchers in selecting the most appropriate approach for their specific task and data in IE. Furthermore, our study highlights recent trends that have the potential to expand the scope and applicability of OIE, creating promising avenues for future research in this area. Overall, our aim is that this study will assist new researchers in developing a comprehensive understanding of OIE solutions, and inspire further advances in this field. By providing insights and guidance to the NLP community, we hope to facilitate ongoing research and development in Information Extraction and related areas.

Neural network design implications in OIE — The research presented in Chapter 3 has two primary objectives. First, it investigates the impact of contextualized embeddings in OIE. Second, it examines the performance of different neural network architectures and highlights the superiority of Bi-GRU networks, which are computationally efficient and faster to train. As previously explained, our findings demonstrate the superior performance of ELMo embeddings compared to other textual representations, which can assist researchers in selecting appropriate embeddings for their specific task. Additionally, our results underscore how GRU networks outperform state-of-the-art solutions by capturing complex and long-range dependencies between words in a sentence, making them well-suited for sequence labeling tasks. Furthermore, our study suggests that formulating the OIE task as a sequence tagging problem is a promising research direction, this is consistent with previous research as explained in Chapter 3. As such, our study can provide a useful guideline for researchers considering this formulation when developing their OIE models. Overall, our research highlights the importance of neural network design in OIE and provides insights into the optimal selection of embeddings and neural network architectures.

OIE as a pre-training step — Chapter 4 presents the findings on utilizing OIE as a source task to address the challenge of insufficient training data for neural models. Our experiments on transfer learning demonstrate that OIE can be effectively transferred to semantically related tasks, such as relation extraction, and adapt to new domains, namely biomedical. This highlights the potential of OIE as a promising approach for developing domain-agnostic models. With the success of transfer learning of OIE, we encourage researchers to further explore the potential of OIE in other NLP domains and examine its transferability beyond semantically related tasks, ultimately contributing to the development of more accurate and efficient NLP models.

Challenges in fine-tuning Pre-trained Language Models (PLMs) — Chap-

ter 5 provides important insights into the limitations of language models and the challenges associated with fine-tuning them on downstream tasks. Our extensive experimentation reveals several key findings. Firstly, PLMs struggle to generalize well when the data available for fine-tuning differs from the test data, which we addressed through the use of data augmentation techniques. Secondly, PLMs can suffer from fine-tuning instability, which we mitigated to a large extent through our proposed two-stage fine-tuning model. Thirdly, for the regression subtask, PLMs were found to be unreliable, highlighting the need for alternative approaches. These findings are of vital importance to the NLP research community, particularly as PLMs are increasingly used in a variety of tasks. Our work emphasizes the challenges associated with fine-tuning PLMs on downstream tasks, specifically when there is a scarcity of annotated data, which can limit their generalization. We believe that our work offers valuable guidelines on the use of data augmentation and two-stage fine-tuning to enhance the robustness of language models. Furthermore, our study highlights the potential of PLMs in taxonomy classification and regression tasks but also underscores the need for further research to improve their performance and generalizability.

Attention-based model for improved OIE — Our work in Chapter 6 highlights the potential of adding an attention component to OIE, which allows the model to selectively focus on relevant parts of the input sequence when extracting information. Our findings indicate that self-attention is particularly effective in OIE, as it enables the model to efficiently capture long-range dependencies in the input. As a result, our research shows that self-attention can be a valuable tool for capturing the relationship between words in a sentence, and thus provides a promising starting point for further research in this area. We hope that our work inspires researchers to explore the use of attention mechanisms in OIE and other NLP tasks, as a means of improving the performance and interpretability of these models.

Revolutionizing knowledge representation with OIE — Our final finding in Chapter 6 demonstrates that OIE can significantly aid in the construction of queryable KGs. The pipeline we presented demonstrated how the extracted triples can be used to populate KGs in a fully-automated manner. In addition, the proposed methodology enabled us to evaluate each phase separately in an efficient way, which can serve as a partial alternative to manual validation, particularly for large-scale KGs. Overall, our findings suggest that OIE has a high potential to revolutionize knowledge representation in NLP and should be further explored by researchers in the field. We hope that our work will inspire further research and development in this area, leading to improved knowledge representation and more efficient information retrieval in NLP.

7.3 Research Validity and Limitations

This section discusses the validity and limitations of the research in this dissertation. First, we will restate some of the threats to validity associated with the data used in the OIE model, and then we will discuss the validity of our evaluation of pre-trained language models and knowledge graphs.

Language bias — Throughout this dissertation, we have exploited textual data in our proposed OIE models that were primarily based on the English language. The desired labeled OIE datasets follow the Beginning, Inside, and Outside (BIO) labeling scheme to recognize entity boundaries. BIO tagging is a commonly used format for tagging tokens in a chunking task in computational linguistics. In addition, following the works of Reimers and Gurevych (2017b) and Mozharova and Loukachevitch (2016) that demonstrate the efficiency of the BIO tagging scheme in CRF-based NER models, we opted to use a BIO-tagged corpus in our proposed NER model that was deployed in Open-CyKG in Chapter 6. Obtaining a BIO-annotated labeled dataset in either task was a challenge. Consequently, we were forced to test our models only using the English language. This limited our ability to explore the effect of our proposed models and validate it in different languages for the benefit of the research community.

Data size bias — Following the above-mentioned point, while developing the OIE models, we only had access to a limited amount of BIO-annotated data. Accordingly, we were only able to evaluate the performance our proposed OIE models in Chapters: 3, 4, and 6 on a restricted dataset dimensionality. In Chapter 3 we had access to two BIO-labelled OIE datasets, Newswire and WikiNews, containing 1241 and 1957 sentences respectively, which constituted our training, validation, and development sets. The same WikiNews dataset was utilized in Chapter 4 to investigate the transferability of the OIE system. While in Chapter 6, the OIE model employed in Open-CyKG utilized the MalwareDB dataset, which is composed of 6,819 sentences. However, out of the 6,819 sentences, only 1,910 sentences were classified as informative sentences that contained at least two entities and a relationship label, which challengingly formed our training, test, and validation sets. Having relatively small training and validation sets might lead to imprecise estimation of the performance of our OIE models. Going forward, it is imperative to evaluate the OIE model performance at a larger scale with larger datasets and for other languages as well in order to obtain a better and more comprehensive interpretation of the deployment of OIE in knowledge representation. We anticipate that training our proposed OIE models on more data samples would result in

a more robust performance.

Language model instability — As mentioned in Chapter 5, pre-trained language models suffer from instability during the fine-tuning phase. The instability phenomenon is highly correlated with the random seeds and thus might lead to highly varying results. This phenomenon was observed in our work in Chapter 5. A mitigation technique is to perform several fine-tuning runs with distinct seeds and select the best one from the hold-out. However, this method may result in overfitting on the hold-out set (Mosbach et al., 2021). Additionally, Zhang et al. (2021) and Mosbach et al. (2021) argue that training for a large number of epochs may reduce the instability effect regardless of the chosen seed, as it only might take longer to be able to converge to the same global potential minimum that can be achieved from the best possible seed. The fine-tuning instability hinders the reproducibility of our reported scores in Chapter 5, however, our results analysis is deemed valid as they were based on extensive experimentations with a minimum of ten runs per experiment.

Data bias — During the development of our UU-Tax model in Chapter 5 we relied on data augmentation for our proposed two-stage fine-tuning model for the taxonomy classification sub-task. We mainly used two data augmentation techniques. First, back-translation. Second, Insertion and Substitution augmentation techniques are provided by the NLPAug tool that is based on BERT. We explained in Chapter 5 that the Substitution operation was only performed on sentences that were classified to hold a valid taxonomic relationship —- sentences labeled 1 —. As performing the Substitution operation with sentences originally invalid sentences —- labeled 0 — might result in an inversion to a valid sentence with an invalid label, such a case would confuse the model and have an undesired impact on the results obtained. As a consequence, the data that was used to fine-tune the pre-trained language model became somewhat imbalanced. Precisely, both operations of the NLPAug tool resulted in a ratio of positive to negative samples of 17:8 in all three languages English, French, and Italian. Although the over-representation of positive samples might not have a negative impact on the achieved F-score, the generation of more balanced data is left to important future work.

Expert validation — In Chapter 6 we mentioned that KG validation is an expensive process to be carried out manually. Nevertheless, noisy triples are inevitably introduced into KGs that could be produced by automatic extraction of triples from large-scale unstructured text. As a result, this raises the risk that our KG may not yet represent all potentially relevant input. To validate our KG, we opted to validate each of the three components that constitute Open-CyKG separately, the proposed attention-OIE model, the Bi-GRU-CRF NER model, and finally, fusion and canonicalization techniques using HAC.

Alternatively, it would be valuable for domain experts to validate the content of the KG through expert evaluations to ensure that information enclosed and fused is both relative and informative to cybersecurity analysts.

7.4 Future Research

This dissertation has contributed to addressing the ongoing challenges in the development of IE techniques and their deployment to formulate an efficient representation of the extracted textual knowledge. Nonetheless, IE remains an active research topic with several unanswered questions. Throughout the dissertation, we have investigated numerous ways to build OIE systems and demonstrated their efficiency in knowledge representation by building an end-to-end pipeline from unstructured textual data to create a queryable knowledge graph. This opens the door for practical and promising research directions to develop more accurate IE methodologies to contribute toward an efficient and effective data representation. In all of these areas, there is more work that can be done to benefit knowledge representation, we discuss the following selection of particularly interesting topics: beyond binary triples, dealing with instability in pre-trained language models, knowledge graph validation, and ways to further improve knowledge representations.

Beyond binary triples — One of the limitations of our proposed OIE models in this dissertation is that we limited the extraction to binary relations. Dual extractions might not represent information adequately, as the extracted fact might be correct but incomplete. This leads to a negative effect on several downstream OIE applications, while n-ary tuples[1] retain information held in text. Additionally, we could benefit from nested tuples. For instance the sentence; *'The senator approved the deal to build a monorail.'* nested extraction would result in the following $\ll$ *the senator, approved,* $\ll$ *the deal, to build, monorail* $\gg\gg$. Albeit being more complex to extract and error-prone, n-ary extractions and nested extractions open the door for fact completeness that results in higher quality extractions. Even after potential implementation, further research should investigate ways to deploy such extractions efficiently in downstream applications to ensure that they are informative, for instance, integrating rule-based linguistic methods to alleviate erroneous and uninformative extractions and explore whether such extractions provide benefit in practice.

Pre-trained language model instability — Language models, specifically the pre-training-then-fine-tuning paradigm, have successfully contributed to many

[1] n-ary is defined as facts that enclose more than two arguments.

NLP tasks. Nonetheless, overfitting hinders the fine-tuning process, which results in the model failing to generalize well on the downstream task during inference. This phenomenon was observed in our work in UU-TAX methodology for taxonomy classification using a two-stage process for fine-tuning pre-trained language models as defined in Chapter 5. We aimed to improve the model's generalizability using data augmentation techniques by applying two fine-tuning stages. However, this was not enough to fully address the instability matter. Previous work has investigated this drawback, but most of the previous research focuses solely on BERT (Mosbach et al., 2021; Devlin et al., 2019b). Our aim in the future is to focus on discriminative language models such as ELECTRA, and we briefly investigated the prompt-based learning effect on the ELECTRA model in Chapter 5. Nonetheless, in the future, we want to dive deeper into investigating mitigation techniques for the instability associated with discriminative models.

Knowledge graph validity — Despite our efforts in validating the KG described in Chapter 6 by validating each component of our proposed Open-CyKG methodology separately, KGs inevitably hold unfitting or incorrect triples, which negatively affect their integration in downstream applications. Another approach worth investigating is using automated techniques without the need for any human involvement to validate the KG by measuring the degree to which the enclosed triples in the KG are semantically correct. A simple and efficient methodology to achieve this is by assigning a confidence score to the KG triples based on comparing the contained knowledge to external sources and later eliminating low-weighted instances. As promising as this approach sounds, it would not efficiently measure the coverage of the KG. In our proposed Open-CyKG methodology, we were able to determine the coverage of information enclosed in the original unstructured document that was used to populate the KG by evaluating the OIE component. In addition, the automatic validation approach opens the room to further questions such as: "How can we define an appropriate knowledge source for comparison?" and "Is this solution scalable to large KGs with dynamic data?". Therefore, validating the correctness of KGs remains an open challenge and a vital future research direction.

Towards a better knowledge representation — At the start of this research, our long-term ambition was to investigate ways to build an efficient knowledge representation, this was achieved in our proposed Open-CyKG model in Chapter 6 by primarily relying on OIE to populate the KG and representing the data in the form of $\ll$ *head, relationship, tail* $\gg$. Nonetheless, the KG constructed in Open-CyKG was relatively small, this representation may not be scalable when it comes to large KGs. Additionally, the traditional representation does

not support tasks such as link and entity prediction. Thus, inspired by word embeddings, distributed representation was later used to represent the information enclosed within a KG by assuming that both the entities and relations are vectors in the semantic space. One of the early models used to represent one-to-one relationships is TransE (Translating Embeddings for Modeling Multi-relational Data) (Bordes et al., 2013) which later evolved to TransH to represent many-to-many relations by interpreting a relation as a translation process on a hyperplane (Wang et al., 2014). In the future, we would like to investigate ways to contribute toward an even more competent and scalable representation, specifically entity-agnostic KGs as it would alleviate the need to represent all entities and relational triples in KGs, this is deemed important in downstream tasks such as knowledge-grounded conversation generation.

7.5 Personal Reflection

When I first decided to start my Ph.D., I had only three things running through my mind: The reason I was doing this, the problem I was trying to tackle, and how my work could be different from other researchers. In these coming paragraphs, I will put my thoughts into words on the effect it had on me, both, personally and professionally.

Uncovering the passion for knowledge — In general, a Ph.D. is about more than just the topic of choice, it is rather about growing as a researcher. It is deemed like a marathon where things often seem more urgent than they are. I have learned not to act immediately on the emotions the process elicits but to calmly and thoroughly think about my state in relation to my research. It was essential to always keep in mind the motive behind my chosen Ph.D. subject. The reason behind my choice of diving deeper into Information Extraction and Knowledge Representation, and in particular Open Information Extraction, is that I was intrigued by endless questions that ran through my mind whilst coming across so many NLP topics. After conducting a literature review on Information Extraction that resulted in my dissatisfaction from leaving my questions unanswered, I was eager to explore and dive deeply into the subject. This challenged me to pursue the matter further and quench my thirst for unearthing the information I needed. Passion was the primary key to my dedication and hard work. Nevertheless, I have yet to uncover a lot more. I believe that my research and involvement in Information Extraction are far from done. I view the Ph.D. as just the beginning rather than the end goal.

Remaining vigilant and facing the challenge of change — It was not always a linear road when it came to the process of finishing my Ph.D... setbacks and

desperation sometimes got the better of me. It was an actual roller coaster of emotions, the intense highs and lows, moments of great accomplishments and great sadness, and feeling on top of schedule yet stretched across a hundred tasks. It was not only about how far I had gone academically but personally as well. It was critical to allocate time for thinking beyond the scope of work and coming to peace with the fact that working does not usually entail producing. After all, the greater the obstacle, the more glory in overcoming it. That made room for a flexible mental state that later contributed novel ideas and recharged my passion. A Ph.D. is not entirely about how we advance in a particular subject but rather about our ability to navigate the journey and how it helps us evolve. It was improbable to anticipate how my Ph.D. storyline would unravel, but that is what made the journey exhilarating. Taking it one step at a time, adapting to changes, and coming out the other end of failures with a true win was my leading guide to the completion of my dissertation.

Knowledge will bring you the opportunity to make a difference — I have always envisioned myself building something that could be beneficial to others. I consider this partially achieved with Open-CyKG. When I first started working on the Open-CyKG model, to ultimately reach the goal of this dissertation, I made a list of everything I wanted to cover. Then I came to the realization that one can not possibly do everything, because I will never be fully satisfied with the outcome, I will always want more. When Open-CyKG came to light, and the paper was published with positive reviews advancing me to improve the model even further, all moments of hardships vanished. What was absolutely heart-warming was the feedback I got from multiple researchers showing interest in the model we built and being informed that it is currently used as teaching material in faculty courses. I came to the realization that I would like to follow through and continue contributing time, sources, and energy to open-science. Although academic life is often measured by the number of citations you have, this was not my vision, my ultimate goal was to feel the impact of my research and working toward the greater good, and I am beyond grateful that I had that feeling bestowed upon me during my Ph.D. journey.
There are so many more complex socio-economic aspects that can drastically change one's experience in their journey to finalizing their Ph.D. Feeling burnout in the middle of this long and time-demanding process, all the while combating imposter feelings, could be a considerable burden. Trying to accomplish realistic goals in a specific timeframe, running towards the dream of becoming a Ph.D. holder, and never stopping to catch my breath only to be surprised by all the unexpected accomplishments that built me into the researcher I am today.

Bibliography

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D. G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., Wicke, M., Yu, Y., and Zheng, X. (2016). Tensorflow: A system for large-scale machine learning. In *Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation*, OSDI'16, page 265–283, USA. USENIX Association.

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

Akbik, A. and Löser, A. (2012). Kraken: N-ary facts in open information extraction. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction (AKBC-WEKEX)*, pages 52–56.

Altiti, O., Abdullah, M., and Obiedat, R. (2020). Just at semeval-2020 task 11: Detecting propaganda techniques using bert pre-trained model. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1749–1755.

Amit, S. (2012). Introducing the knowledge graph: Things, not strings. *Official Blog (of Google), 2012*.

Anaby-Tavor, A., Carmeli, B., Goldbraich, E., Kantor, A., Kour, G., Shlomov, S., Tepper, N., and Zwerdling, N. (2020). Do not have enough data? deep learning to the rescue! In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7383–7390.

Arslan, M. and Cruz, C. (2022). Semantic enrichment of taxonomy for bi applications using multifaceted data sources through nlp techniques. *Procedia Computer Science*, 207:2424–2433.

Awad, M. and Khanna, R. (2015). Support vector regression. In *Efficient learning machines*, pages 67–80. Springer.

Ayranci, P., Lai, P., Phan, N., Hu, H., Kolinowski, A., Newman, D., and Dou, D. (2022). Onml: an ontology-based approach for interpretable machine learning. *Journal of Combinatorial Optimization*, pages 1–24.

Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. *ICLR*.

Banko, M. and Etzioni, O. (2008). The tradeoffs between open and traditional relation extraction. In *Proceedings of ACL-08: HLT*, pages 28–36.

Bansal, T., Belanger, D., and McCallum, A. (2016). Ask the gru: Multi-task learning for deep text recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems*, RecSys '16, page 107–114, New York, NY, USA. Association for Computing Machinery.

Bao, S., He, H., Wang, F., Wu, H., and Wang, H. (2020). Plato: Pre-trained dialogue generation model with discrete latent variable. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 85–96.

Bast, H. and Haussmann, E. (2013). Open information extraction via contextual sentence decomposition. In *2013 IEEE Seventh International Conference on Semantic Computing*, pages 154–159. IEEE.

Beckerman, J. and Christakis, T. (2019). Learning open information extraction of implicit relations from reading comprehension datasets.

Bengio, Y. (2012). Practical recommendations for gradient-based training of deep architectures. In *Neural networks: Tricks of the trade*, pages 437–478. Springer.

Bengio, Y., Ducharme, R., and Vincent, P. (2001). A neural probabilistic language model. *Advances in neural information processing systems*, 13.

Bhatia, P., Arumae, K., and Celikkaya, B. (2020). Dynamic transfer learning for named entity recognition. In *Precision Health and Medicine*.

Bird, S. (2006). NLTK: the natural language toolkit. In *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*, pages 69–72.

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J. (2008). Freebase: A collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250.

Bordea, G., Lefever, E., and Buitelaar, P. (2016). Semeval-2016 task 13: Taxonomy extraction evaluation (texeval-2). In *Proceedings of the 10th international workshop on semantic evaluation (semeval-2016)*, pages 1081–1091.

Bordes, A., Chopra, S., and Weston, J. (2014). Question answering with subgraph embeddings. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 615–620.

Bordes, A. and Gabrilovich, E. (2014). Constructing and mining web-scale knowledge graphs: KDD 2014 tutorial. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1967–1967.

Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., and Yakhnenko, O. (2013). Translating embeddings for modeling multi-relational data. In Burges, C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K., editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc.

Boubekeur, F. and Azzoug, W. (2013). Concept-based indexing in text information retrieval. *arXiv preprint arXiv:1303.1703*.

Bridges, R. A., Jones, C. L., Iannacone, M. D., and Goodall, J. R. (2013). Automatic labeling for entity extraction in cyber security. *ArXiv*, abs/1308.4941.

Brin, S. (1999). Extracting patterns and relations from the world wide web. In Atzeni, P., Mendelzon, A., and Mecca, G., editors, *The World Wide Web and Databases*, pages 172–183, Berlin, Heidelberg. Springer Berlin Heidelberg.

Brunato, D., Chesi, C., Chowdhury, S. A., Dell'Orletta, F., Montemagni, S., Venturi, G., and Zamparelli, R. (2022). Presupposed taxonomies: Evaluating neural network semantics (pretens). `https://sites.google.com/view/semeval2022-pretens/`, `https://github.com/shammur/SemEval2022Task3`.

Cabral, B. S., Glauber, R., Souza, M., and Claro, D. B. (2020). CrossOIE: cross-lingual classifier for open information extraction. In *International Conference on Computational Processing of the Portuguese Language*, pages 368–378. Springer.

Ceccarelli, D., Lucchese, C., Orlando, S., Perego, R., and Trani, S. (2013). Dexter: an open source framework for entity linking. In *Proceedings of the sixth international workshop on Exploiting semantic annotations in information retrieval*, pages 17–20.

Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.

Chollet, F. (2015). Keras. `https:/https://github.com/fchollet/keras/`.

Christensen, J., Mausam, Soderland, S., and Etzioni, O. (2011a). An analysis of open information extraction based on semantic role labeling. In *Proceedings of the Sixth International Conference on Knowledge Capture*, K-CAP '11, page 113–120, New York, NY, USA. Association for Computing Machinery.

Christensen, J., Mausam, Soderland, S., and Etzioni, O. (2011b). An analysis of open information extraction based on semantic role labeling. In *K-CAP '11*.

Christensen, J., Soderland, S., Etzioni, O., et al. (2010). Semantic role labeling for open information extraction. In *Proceedings of the NAACL HLT 2010 first international workshop on formalisms and methodology for learning by reading*, pages 52–60.

Chung, H., Iorga, M., Voas, J., and Lee, S. (2017). "alexa, can i trust you?". *Computer*, 50(9):100–104.

Chung, J., Gülçehre, Ç., Cho, K., and Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. *CoRR*, abs/1412.3555.

Clark, K., Luong, M.-T., Le, Q. V., and Manning, C. D. (2020). Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*.

Collobert, R. and Weston, J. (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167.

Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, É., Ott, M., Zettlemoyer, L., and Stoyanov, V. (2020a). Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451.

Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., and Stoyanov, V. (2020b). Unsupervised cross-lingual representation learning at scale. In *ACL*.

Cui, L., Wei, F., and Zhou, M. (2018). Neural open information extraction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 407–413.

Davis, E. (2015). Knowledge representation. In Wright, J. D., editor, *International Encyclopedia of the Social Behavioral Sciences (Second Edition)*, pages 98–104. Elsevier, Oxford, second edition edition.

Del Corro, L. and Gemulla, R. (2013). Clausie: clause-based open information extraction. In *Proceedings of the 22nd international conference on World Wide Web*, pages 355–366.

Delen, D. (2014). *Real-world data mining: applied business analytics and decision making*. FT Press.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019a). Bert: Pre-training of deep bidirectional transformers for language understanding.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019b). BERT: Pre-training of deep bidirectional transformers for language understanding. 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Dey, R. and Salem, F. M. (2017). Gate-variants of gated recurrent unit (gru) neural networks. In *2017 IEEE 60th international midwest symposium on circuits and systems (MWSCAS)*, pages 1597–1600. IEEE.

Do, C. B. and Ng, A. Y. (2005). Transfer learning for text classification. In Weiss, Y., Schölkopf, B., and Platt, J., editors, *Advances in Neural Information Processing Systems*, volume 18. MIT Press.

Dong, X. L., Gabrilovich, E., Heitz, G., Horn, W., Lao, N., Murphy, K., Strohmann, T., Sun, S., and Zhang, W. (2014). Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, pages 601–610. Evgeniy Gabrilovich Wilko Horn Ni Lao Kevin Murphy Thomas Strohmann Shaohua Sun Wei Zhang Geremy Heitz.

Egozi, O., Markovitch, S., and Gabrilovich, E. (2011). Concept-based information retrieval using explicit semantic analysis. *ACM Transactions on Information Systems (TOIS)*, 29(2):1–34.

Etzioni, O., Banko, M., Soderland, S., and Weld, D. S. (2008). Open information extraction from the web. *Communications of the ACM*, 51(12):68–74.

Etzioni, O., Cafarella, M., Downey, D., Popescu, A.-M., Shaked, T., Soderland, S., Weld, D. S., and Yates, A. (2005). Unsupervised named-entity extraction from the web: An experimental study. *ARTIFICIAL INTELLIGENCE*, 165:91–134.

Etzioni, O., Fader, A., Christensen, J., Soderland, S., et al. (2011). Open information extraction: The second generation. In *Twenty-Second International Joint Conference on Artificial Intelligence*.

Fabian, M., Gjergji, K., Gerhard, W., et al. (2007). Yago: A core of semantic knowledge unifying wordnet and wikipedia. In *16th International World Wide Web Conference, WWW*, pages 697–706.

Fader, A., Soderland, S., and Etzioni, O. (2011). Identifying relations for open information extraction. In *Proceedings of the 2011 conference on empirical methods in natural language processing*, pages 1535–1545.

Fan, A., Lavril, T., Grave, E., Joulin, A., and Sukhbaatar, S. (2020). Addressing some limitations of transformers with feedback memory. *arXiv preprint arXiv:2002.09402, 2020*.

Fei, H., Ren, Y., Zhang, Y., Ji, D., and Liang, X. (2020). Enriching contextualized language model from knowledge graph for biomedical information extraction. *Briefings in Bioinformatics, 2020*.

Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. Bradford Books.

Fensel, D., Şimşek, U., Angele, K., Huaman, E., Kärle, E., Panasiuk, O., Toma, I., Umbrich, J., and Wahler, A. (2020). *Knowledge Graphs*. Springer.

Ferrucci, D. A. (2012). Introduction to "this is watson". *IBM Journal of Research and Development*, 56(3.4):1:1–1:15.

Finkel, J. R., Grenager, T., and Manning, C. D. (2005). Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 363–370.

Francis, N., Green, A., Guagliardo, P., Libkin, L., Lindaaker, T., Marsault, V., Plantikow, S., Rydberg, M., Selmer, P., and Taylor, A. (2018). Cypher: An evolving query language for property graphs. In *Proceedings of the 2018 International Conference on Management of Data*, pages 1433–1445.

Galárraga, L., Heitz, G., Murphy, K., and Suchanek, F. M. (2014). Canonicalizing open knowledge bases. In *Proceedings of the 23rd acm international conference on conference on information and knowledge management*, pages 1679–1688.

Gamallo, P. (2014). An Overview of Open Information Extraction (Invited talk). In Pereira, M. J. V., Leal, J. P., and Simões, A., editors, *3rd Symposium on Languages, Applications and Technologies*, volume 38 of *OpenAccess Series in Informatics (OASIcs)*, pages 13–16, Dagstuhl, Germany. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

Gamallo, P., Garcia, M., and Fernández-Lanza, S. (2012). Dependency-based open information extraction. In *Proceedings of the joint workshop on unsupervised and semi-supervised learning in NLP*, pages 10–18.

Gao, S., Young, M. T., Qiu, J. X., Yoon, H.-J., Christian, J. B., Fearn, P. A., Tourassi, G. D., and Ramanthan, A. (2018). Hierarchical attention networks for information extraction from cancer pathology reports. *Journal of the American Medical Informatics Association*, 25(3):321–330.

Gasmi, H., Laval, J., and Bouras, A. (2019). Information extraction of cybersecurity concepts: An LSTM approach. *Applied Sciences*, 9(19):3945.

Gehring, J., Auli, M., Grangier, D., Yarats, D., and Dauphin, Y. N. (2017). Convolutional sequence to sequence learning. In *International conference on machine learning*, pages 1243–1252. PMLR.

Georgescu, T.-M., Iancu, B., Zamfiroiu, A., Doinea, M., Boja, C. E., and Cartas, C. (2021). A survey on named entity recognition solutions applied for cybersecurity-related text processing. In *Proceedings of Fifth International Congress on Information and Communication Technology*, pages 316–325. Springer.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, 27.

Goodman, A., Pepe, A., Blocker, A. W., Borgman, C. L., Cranmer, K., Crosas, M., Di Stefano, R., Gil, Y., Groth, P., Hedstrom, M., et al. (2014). Ten simple rules for the care and feeding of scientific data. *PLoS computational biology*, 10(4):e1003542.

Graves, A., Mohamed, A.-r., and Hinton, G. (2013). Speech recognition with deep recurrent neural networks. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 6645–6649.

Hachey, B., Radford, W., Nothman, J., Honnibal, M., and Curran, J. R. (2013). Evaluating entity linking with wikipedia. *Artificial intelligence*, 194:130–150.

Halpern, J. Y. (2017). *Reasoning about uncertainty*. MIT press.

Han, J. and Wang, H. (2021). Transformer based network for open information extraction. *Engineering Applications of Artificial Intelligence*, 102:104262.

He, L., Lewis, M., and Zettlemoyer, L. (2015). Question-answer driven semantic role labeling: Using natural language to annotate natural language. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 643–653, Lisbon, Portugal. Association for Computational Linguistics.

Hendrickx, I., Kim, S. N., Kozareva, Z., Nakov, P., Séaghdha, D. , Padó, S., Pennacchiotti, M., Romano, L., and Szpakowicz, S. (2009). Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *SEW@NAACL-HLT*, pages 94–99.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Hoffmann, R., Zhang, C., Ling, X., Zettlemoyer, L., and Weld, D. S. (2011). Knowledge-based weak supervision for information extraction of overlapping relations. In *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies*, pages 541–550.

Howard, J. and Ruder, S. (2018). Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. Association for Computational Linguistics.

Hu, J., Niu, H., Carrasco, J., Lennox, B., and Arvin, F. (2020). Voronoi-based multi-robot autonomous exploration in unknown environments via deep reinforcement learning. *IEEE Transactions on Vehicular Technology*, 69(12):14413–14423.

Huang, H., Heck, L. P., and Ji, H. (2015). Leveraging deep neural networks and knowledge graphs for entity disambiguation. *CoRR*, abs/1504.07678, 2015.

Huang, S., Luo, X., Huang, J., Guo, Y., and Gu, S. (2019). An unsupervised approach for learning a chinese is-a taxonomy from an unstructured corpus. *Knowledge-Based Systems*, 182:104861.

Jain, M., Rogers, R. D., and Librarian, F. Y. (1985). "we are drowning in information and starving for knowledge.".

Jones, C. L., Bridges, R. A., Huffer, K. M., and Goodall, J. R. (2015). Towards a relation extraction framework for cyber-security concepts. In *Proceedings of the 10th Annual Cyber and Information Security Research Conference*, pages 1–4.

Junge, A. and Jensen, L. J. (2020). Cocoscore: context-aware co-occurrence scoring for text mining applications using distant supervision. *Bioinformatics*, 36(1):264–271.

Kaiser, . and Sutskever, I. (2015). Neural gpus learn algorithms. *arXiv preprint arXiv:1511.08228*.

Kannan, A., Kurach, K., Ravi, S., Kaufmann, T., Tomkins, A., Miklos, B., Corrado, G., Lukacs, L., Ganea, M., Young, P., and Ramavajjala, V. (2016). Smart reply: Automated response suggestion for email. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 955–964, New York, NY, USA. Association for Computing Machinery.

Keneshloo, Y., Ramakrishnan, N., and Reddy, C. K. (2019). Deep transfer reinforcement learning for text summarization. In *SDM*.

Kiesling, E., Ekelhart, A., Kurniawan, K., and Ekaputra, F. (2019). The SEPSES knowledge graph: an integrated resource for cybersecurity. In *International Semantic Web Conference*, pages 198–214. Springer.

Kim, G., Lee, C., Jo, J., and Lim, H. (2020). Automatic extraction of named entities of cyber threats using a deep Bi-LSTM-CRF network. *International Journal of Machine Learning and Cybernetics*, 11(10):2341–2355.

Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. In *In 3rd International Conference on Learning Representations, ICLR, San Diego, CA, USA, Conference Track Proceedings*.

Kneser, R. and Ney, H. (1995). Improved backing-off for m-gram language modeling. *1995 International Conference on Acoustics, Speech, and Signal Processing*, 1:181–184 vol.1.

Kocijan, V., Cretu, A.-M., Camburu, O.-M., Yordanov, Y., and Lukasiewicz, T. (2019). A surprisingly robust trick for the winograd schema challenge. In Korhonen, A. and Traum, D., editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28 - August 2, 2019*. Association for Computational Linguistics.

Kramer, O. (2013). K-nearest neighbors. In *Dimensionality reduction with unsupervised nearest neighbors*, pages 13–23. Springer.

Kumar, V., Choudhary, A., and Cho, E. (2020). Data augmentation using pre-trained transformer models. In *Proceedings of the 2nd Workshop on Lifelong Learning for Spoken Language Systems*, pages 18–26, Suzhou, China. Association for Computational Linguistics.

Kuru, O., Can, O. A., and Yuret, D. (2016). Charner: Character-level named entity recognition. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 911–921.

Lavine, B. and Blank, T. (2009). 3.18-feed-forward neural networks, in comprehensive chemometrics, edited by steven d. brown, romá tauler and beata walczak.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436–444.

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

Li, B. and Rudzicz, F. (2021). Torontocl at cmcl 2021 shared task: Roberta with multi-stage fine-tuning for eye-tracking prediction. In *CMCL at NAACL*.

Li, B., Sainath, T. N., Narayanan, A., Caroselli, J., Bacchiani, M., Misra, A., Shafran, I., Sak, H., Pundak, G., Chin, K. K., et al. (2017a). Acoustic modeling for google home. In *Interspeech*, pages 399–403.

Li, P.-H., Dong, R.-P., Wang, Y.-S., Chou, J.-C., and Ma, W.-Y. (2017b). Leveraging linguistic structures for named entity recognition with bidirectional recursive neural networks. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2664–2669, Copenhagen, Denmark. Association for Computational Linguistics.

Lim, S. K., Muis, A. O., Lu, W., and Ong, C. H. (2017). Malwaretextdb: A database for annotated malware articles. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1557–1567.

Lin, H., Yan, J., Qu, M., and Ren, X. (2019). Learning dual retrieval module for semi-supervised relation extraction. In *The World Wide Web Conference*, WWW '19, page 1073–1083, New York, NY, USA. Association for Computing Machinery.

Lin, T., Mausam, and Etzioni, O. (2012). Entity linking at web scale. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction (AKBC-WEKEX)*, pages 84–88, Montréal, Canada. Association for Computational Linguistics.

Liu, G.-H., Yang, J.-Y., and Li, Z. (2015). Content-based image retrieval using computational visual attention model. *pattern recognition*, 48(8):2554–2566.

Liu, Q., Kusner, M. J., and Blunsom, P. (2020a). A survey on contextual embeddings. *arXiv preprint arXiv:2003.07278*.

Liu, W., Zhou, P., Zhao, Z., Wang, Z., Ju, Q., Deng, H., and Wang, P. (2020b). K-bert: Enabling language representation with knowledge graph. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 2901–2908.

Liu, X., Cheng, H., He, P., Chen, W., Wang, Y., Poon, H., and Gao, J. (2020c). Adversarial training for large neural language models. *arXiv preprint arXiv:2004.08994*.

Loshchilov, I. and Hutter, F. (2019). Decoupled weight decay regularization. In *ICLR*.

Luo, L., Yang, Z., Yang, P., Zhang, Y., Wang, L., Lin, H., and Wang, J. (2018). An attention-based bilstm-crf approach to document-level chemical named entity recognition. *Bioinformatics*, 34(8):1381–1388.

Luu, A. T., Tay, Y., Hui, S. C., and Ng, S. K. (2016). Learning term embeddings for taxonomic relation identification using dynamic weighting neural network. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 403–413.

Ma, E. (2019). Nlp augmentation. https://github.com/makcedward/nlpaug.

Ma, X. and Hovy, E. (2016). End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1064–1074, Berlin, Germany. Association for Computational Linguistics.

Mahabadi, R. K., Ruder, S., Dehghani, M., and Henderson, J. (2021). Parameter-efficient multi-task fine-tuning for transformers via shared hypernetworks. In *ACL/IJCNLP*.

Mausam, Schmitz, M., Soderland, S., Bart, R., and Etzioni, O. (2012). Open language learning for information extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 523–534, Jeju Island, Korea. Association for Computational Linguistics.

Mausam, M. (2016). Open information extraction systems and downstream applications. In *Proceedings of the twenty-fifth international joint conference on artificial intelligence*, pages 4074–4077.

McMillan, R. (2013). Open threat intelligence. `https://www.gartner.com/doc/2487216/definition-threat-intelligence/`. Online; accessed 19-February-2021.

Meng, R., Zhao, S., Han, S., He, D., Brusilovsky, P., and Chi, Y. (2017). Deep keyphrase generation. In *Proceedings of the 55th Annual Meeting of the*

Association for Computational Linguistics (Volume 1: Long Papers), pages 582–592, Vancouver, Canada. Association for Computational Linguistics.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *Proceedings of Workshop at ICLR*, 2013.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In Burges, C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K., editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc.

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

Miller, J. J. (2013). Graph database applications and concepts with Neo4j. In *Proceedings of the Southern Association for Information Systems Conference, Atlanta, GA, USA*, volume 2324.

Min, S., Seo, M., and Hajishirzi, H. (2017). Question answering through transfer learning from large fine-grained supervision data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 510–517, Vancouver, Canada. Association for Computational Linguistics.

Mintz, M., Bills, S., Snow, R., and Jurafsky, D. (2009). Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011.

Mitchell, T. M. (1997). *Machine learning, International Edition.* McGraw-Hill Series in Computer Science. McGraw-Hill.

Montgomery, D. C., Peck, E. A., and Vining, G. G. (2021). *Introduction to linear regression analysis.* John Wiley & Sons.

Morwal, S., Jahan, N., and Chopra, D. (2012). Named entity recognition using hidden Markov model (HMM). *International Journal on Natural Language Computing (IJNLC)*, 1(4):15–23.

Mosbach, M., Andriushchenko, M., and Klakow, D. (2021). On the stability of fine-tuning BERT: misconceptions, explanations, and strong baselines. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Mozharova, V. A. and Loukachevitch, N. V. (2016). Combining knowledge and crf-based approach to named entity recognition in russian. In *International Conference on Analysis of Images, Social Networks and Texts*, pages 185–195. Springer.

Muhammad, I., Kearney, A., Gamble, C., Coenen, F., and Williamson, P. (2020). Open information extraction for knowledge graph construction. In *International Conference on Database and Expert Systems Applications*, pages 103–113. Springer.

Mulang', I. O., Singh, K., Prabhu, C., Nadgeri, A., Hoffart, J., and Lehmann, J. (2020). Evaluating the impact of knowledge graph context on entity disambiguation models. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 2157–2160.

Myles, A. J., Feudale, R. N., Liu, Y., Woody, N. A., and Brown, S. D. (2004). An introduction to decision tree modeling. *Journal of Chemometrics: A Journal of the Chemometrics Society*, 18(6):275–285.

Nadeau, D. and Sekine, S. (2007). A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26.

Nair, V. and Hinton, G. E. (2010a). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10, page 807–814, Madison, WI, USA. Omnipress.

Nair, V. and Hinton, G. E. (2010b). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10, page 807–814, Madison, WI, USA. Omnipress.

Narayanan, S. N., Ganesan, A., Joshi, K., Oates, T., Joshi, A., and Finin, T. (2018). Early detection of cybersecurity threats using collaborative cognition. In *2018 IEEE 4th international conference on collaboration and internet computing (CIC)*, pages 354–363. IEEE.

Naseem, U., Khushi, M., Reddy, V., Rajendran, S., Razzak, I., and Kim, J. (2021). Bioalbert: A simple and effective pre-trained language model for biomedical named entity recognition. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7. IEEE.

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

Nguyen, H. L., Vu, D. T., and Jung, J. J. (2020). Knowledge graph fusion for smart systems: A survey. *Information Fusion*, 61:56–70.

Niklaus, C., Cetto, M., Freitas, A., and Handschuh, S. (2018a). A survey on open information extraction. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3866–3878, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Niklaus, C., Cetto, M., Freitas, A., and Handschuh, S. (2018b). A survey on open information extraction. *arXiv preprint arXiv:1806.05599*.

Nouvel, D., Antoine, J.-Y., and Friburger, N. (2011). Pattern mining for named entity recognition. In *Language and Technology Conference*, pages 226–237. Springer.

Otter, D., Medina, J. R., and Kalita, J. K. (2021). A survey of the usages of deep learning for natural language processing. *IEEE Transactions on Neural Networks and Learning Systems*, 32:604–624.

Pai, L. (2020). QiaoNing at SemEval-2020 task 4: Commonsense validation and explanation system based on ensemble of language model. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 415–421, Barcelona (online). International Committee for Computational Linguistics.

Pan, S. J. (2020). Transfer learning. *Learning*, 21:1–2.

Panchenko, A., Faralli, S., Ruppert, E., Remus, S., Naets, H., Fairon, C., Ponzetto, S. P., and Biemann, C. (2016). Taxi at semeval-2016 task 13: a taxonomy induction method based on lexico-syntactic patterns, substrings and focused crawling. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 1320–1327.

Pascanu, R., Mikolov, T., and Bengio, Y. (2013). On the difficulty of training recurrent neural networks. In *International conference on machine learning*, pages 1310–1318. PMLR.

Pearl, J. (2014). *Probabilistic reasoning in intelligent systems*, volume 88. Elsevier.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Pellissier Tanon, T., Vrandečić, D., Schaffert, S., Steiner, T., and Pintscher, L. (2016). From freebase to wikidata: The great migration. In *Proceedings of the 25th international conference on world wide web*, pages 1419–1428.

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

Perone, C. S., Silveira, R., and Paula, T. S. (2018). Evaluation of sentence embeddings in downstream and linguistic probing tasks. *ArXiv*, abs/1806.06259.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Piplai, A., Mittal, S., Joshi, A., Finin, T., Holt, J., and Zak, R. (2020). Creating cybersecurity knowledge graphs from malware after action reports. *IEEE Access*, 8:211691–211703.

Plank, B. and Moschitti, A. (2013). Embedding semantic similarity in tree kernels for domain adaptation of relation extraction. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1498–1507.

Pocostales, J. (2016). Nuig-unlp at semeval-2016 task 13: A simple word embedding-based approach for taxonomy extraction. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 1298–1302.

Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al. (2018). Improving language understanding by generative pre-training.

Radhakrishnan, P., Talukdar, P., and Varma, V. (2018). Elden: Improved entity linking using densified knowledge graphs. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1844–1853.

Ramesh Kashyap, A., Mehnaz, L., Malik, B., Waheed, A., Hazarika, D., Kan, M.-Y., and Shah, R. R. (2021). Analyzing the domain robustness of pretrained language models, layer by layer. In *Proceedings of the Second Workshop on Domain Adaptation for NLP*, pages 222–244, Kyiv, Ukraine. Association for Computational Linguistics.

Ramshaw, L. and Marcus, M. (1995). Text chunking using transformation-based learning. In *Third Workshop on Very Large Corpora*.

Ramshaw, L. A. and Marcus, M. P. (1999a). *Text Chunking Using Transformation-Based Learning*, pages 157–176. Springer Netherlands, Dordrecht.

Ramshaw, L. A. and Marcus, M. P. (1999b). Text chunking using transformation-based learning. In *Natural language processing using very large corpora*, pages 157–176. Springer.

Reimers, N. and Gurevych, I. (2017a). Reporting score distributions makes a difference: Performance study of lstm-networks for sequence tagging. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 338–348.

Reimers, N. and Gurevych, I. (2017b). Reporting score distributions makes a difference: Performance study of lstm-networks for sequence tagging. *CoRR*, abs/1707.09861.

Riaño, D., Peleg, M., and ten Teije, A. (2019). Ten years of knowledge representation for health care (2009–2018): Topics, trends, and challenges. *Artificial Intelligence in Medicine*, 100:101713.

Riedel, S., Yao, L., and McCallum, A. (2010). Modeling relations and their mentions without labeled text. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 148–163. Springer.

Roller, S., Kiela, D., and Nickel, M. (2018). Hearst patterns revisited: Automatic hypernym detection from large text corpora. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 358–363, Melbourne, Australia. Association for Computational Linguistics.

Rush, A. M., Chopra, S., and Weston, J. (2015). A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal. Association for Computational Linguistics.

Russo, E. R., Di Sorbo, A., Visaggio, C. A., and Canfora, G. (2019). Summarizing vulnerabilities' descriptions to support experts during vulnerability assessment activities. *Journal of Systems and Software*, 156:84–99.

Sarhan, I., El-Sonbaty, Y., and Abou El-Nasr, M. (2016a). Semi-supervised pattern based algorithm for arabic relation extraction. In *2016 IEEE 28th international conference on tools with artificial intelligence (ICTAI)*, pages 177–183. IEEE.

Sarhan, I., El-Sonbaty, Y., and El-Nasr, M. (2016b). Arabic relation extraction: a survey. *International Journal of Computer and Information Technology*, 5(5).

Sarhan, I., Mosteiro, P., and Spruit, M. (2022). UU-tax at SemEval-2022 task 3: Improving the generalizability of language models for taxonomy classification through data augmentation. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 271–281, Seattle, United States. Association for Computational Linguistics.

Sarhan, I. and Spruit, M. (2018). Uncovering algorithmic approaches in open information extraction: A literature review. In *30th Benelux Conference on Artificial Intelligence*, pages 223–234. Springer CSAI/JADS.

Sarhan, I. and Spruit, M. (2020). Can we survive without labelled data in NLP? Transfer learning for open information extraction. *Applied Sciences*, 10(17):5758.

Sarhan, I. and Spruit, M. (2021). Open-cykg: An open cyber threat intelligence knowledge graph. *Knowledge-Based Systems*, 233:107524.

Sarhan, I. and Spruit, M. R. (2019). Contextualized word embeddings in a neural open information extraction model. In *Natural Language Processing and Information Systems*, pages 359–367, Cham. Springer International Publishing.

Sari, Y., Hassan, M. F., and Zamin, N. (2010). Rule-based pattern extractor and named entity recognition: A hybrid approach. In *2010 International Symposium on Information Technology*, volume 2, pages 563–568. IEEE.

Schmitz, M., Soderland, S., Bart, R., Etzioni, O., et al. (2012). Open language learning for information extraction. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning*, pages 523–534.

Schütze, H., Manning, C. D., and Raghavan, P. (2008). *Introduction to information retrieval*, volume 39. Cambridge University Press Cambridge.

Segura-Bedmar, I., Martinez, P., and de Pablo-Sánchez, C. (2011). Using a shallow linguistic kernel for drug–drug interaction extraction. *Journal of biomedical informatics*, 44(5):789–804.

Simran, K., Sriram, S., Vinayakumar, R., and Soman, K. (2019). Deep learning approach for intelligent named entity recognition of cyber security. In *International Symposium on Signal Processing and Intelligent Recognition Systems*, pages 163–172. Springer.

Smith, E., Papadopoulos, D., Braschler, M., and Stockinger, K. (2022). Lillie: Information extraction and database integration using linguistics and learning-based algorithms. *Information Systems*, 105:101938.

Soderland, S., Roof, B., Qin, B., Xu, S., Mausam, and Etzioni, O. (2010). Adapting open information extraction to domain-specific relations. *AI Mag.*, 31:93–102.

Stanovsky, G. and Dagan, I. (2016). Creating a large benchmark for open information extraction. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2300–2305.

Stanovsky, G., Michael, J., Zettlemoyer, L., and Dagan, I. (2018). Supervised open information extraction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 885–895.

Su, P. and Vijay-Shanker, K. (2020). Adversarial learning for supervised and semi-supervised relation extraction in biomedical literature. *arXiv preprint arXiv:2005.04277*.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27.

Sutton, C. and McCallum, A. (2012). An introduction to conditional random fields. *Foundations and Trends¬Æ in Machine Learning*, 4(4):267–373.

Tappert, C. C. (2019). Who is the father of deep learning? In *2019 International Conference on Computational Science and Computational Intelligence (CSCI)*, pages 343–348. IEEE.

Tawfik, N. S. and Spruit, M. R. (2020). Evaluating sentence representations for biomedical text: Methods and experimental results. *Journal of biomedical informatics*, 104:103396.

Thawani, A., Hu, M., Hu, E., Zafar, H., Divvala, N. T., Singh, A., Qasemi, E., Szekely, P. A., and Pujara, J. (2019). Entity linking to knowledge graphs to infer column types and properties. *SemTab@ ISWC*, 2019:25–32.

Tjong Kim Sang, E. F. and De Meulder, F. (2003). Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Tran, Q., MacKinlay, A., and Jimeno Yepes, A. (2017). Named entity recognition with stack residual LSTM and trainable bias decoding. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 566–575, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Trisedya, B. D., Qi, J., and Zhang, R. (2019). Entity alignment between knowledge graphs using attribute embeddings. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 297–304.

Tu, L., Lalwani, G., Gella, S., and He, H. (2020). An empirical study on robustness to spurious correlations using pre-trained language models. *Transactions of the Association for Computational Linguistics*, 8:621–633.

Turing, A. M. (1950). I.—COMPUTING MACHINERY AND INTELLIGENCE. *Mind*, LIX(236):433–460.

van Haastrecht, M., Sarhan, I., Shojaifar, A., Baumgartner, L., Mallouli, W., and Spruit, M. (2021a). A threat-based cybersecurity risk assessment approach addressing sme needs. In *The 16th International Conference on Availability, Reliability and Security*, pages 1–12.

van Haastrecht, M., Sarhan, I., Yigit Ozkan, B., Brinkhuis, M., and Spruit, M. (2021b). Symbals: A systematic review methodology blending active learning and snowballing. *Frontiers in research metrics and analytics*, 6:33.

Van Harmelen, F., Lifschitz, V., and Porter, B. (2008). *Handbook of knowledge representation*. Elsevier.

Van Le, D., Montgomery, J., Kirkby, K., and Scanlan, J. (2022). Adding an inception network to neural network open information extraction. *IEEE Intelligent Systems*.

Vashishth, S., Jain, P., and Talukdar, P. (2018a). Cesi: Canonicalizing open knowledge bases using embeddings and side information. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, page 1317–1327, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

Vashishth, S., Jain, P., and Talukdar, P. (2018b). Cesi: Canonicalizing open knowledge bases using embeddings and side information. In *Proceedings of the 2018 World Wide Web Conference*, pages 1317–1327.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. (2017). Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Vukotic, V., Raymond, C., and Gravier, G. (2016). A step beyond local observations with a dialog aware bidirectional gru network for spoken language understanding. In *INTERSPEECH*.

Wang, C., Liang, S., Jin, Y., Wang, Y., Zhu, X., and Zhang, Y. (2020). SemEval-2020 task 4: Commonsense validation and explanation. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 307–321, Barcelona (online). International Committee for Computational Linguistics.

Wang, H., Qin, K., Zakari, R. Y., Lu, G., and Yin, J. (2022). Deep neural network-based relation extraction: an overview. *Neural Computing and Applications*, pages 1–21.

Wang, H., Zhao, M., Xie, X., Li, W., and Guo, M. (2019). Knowledge graph convolutional networks for recommender systems. In *The world wide web conference*, pages 3307–3313.

Wang, Z., Zhang, J., Feng, J., and Chen, Z. (2014). Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, AAAI'14, page 1112–1119. AAAI Press.

Welling, M. and Kipf, T. N. (2016). Semi-supervised classification with graph convolutional networks. In *J. International Conference on Learning Representations (ICLR 2017)*.

Wiederhold, G. and McCarthy, J. (1992). Arthur samuel: Pioneer in machine learning. *IBM Journal of Research and Development*, 36(3):329–331.

Wishart, D. S., Knox, C., Guo, A. C., Shrivastava, S., Hassanali, M., Stothard, P., Chang, Z., and Woolsey, J. (2006). Drugbank: a comprehensive resource for in silico drug discovery and exploration. *Nucleic acids research*, 34(suppl_1):D668–D672.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., Drame, M., Lhoest, Q., and Rush, A. (2020). Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Wu, F. and Weld, D. S. (2010). Open information extraction using wikipedia. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 118–127.

Wu, T.-H., Kao, B., Wu, Z., Feng, X., Song, Q., and Chen, C. (2020). Mulce: Multi-level canonicalization with embeddings of open knowledge bases. In Huang, Z., Beek, W., Wang, H., Zhou, R., and Zhang, Y., editors, *Web Information Systems Engineering – WISE 2020*, pages 315–327, Cham. Springer International Publishing.

Xavier, C. C., de Lima, V. L. S., and Souza, M. (2013). Open information extraction based on lexical-syntactic patterns. In *2013 Brazilian Conference on Intelligent Systems*, pages 189–194. IEEE.

Yadav, V. and Bethard, S. (2018). A survey on recent advances in named entity recognition from deep learning models. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2145–2158.

Yang, S., Yu, X., and Zhou, Y. (2020a). Lstm and gru neural network performance comparison study: Taking yelp review dataset as an example. In *2020 International Workshop on Electronic Communication and Artificial Intelligence (IWECAI)*, pages 98–101.

Yang, Y., Cer, D., Ahmad, A., Guo, M., Law, J., Constant, N., Hernandez Abrego, G., Yuan, S., Tar, C., Sung, Y.-h., Strope, B., and Kurzweil, R. (2020b). Multilingual universal sentence encoder for semantic retrieval. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 87–94, Online. Association for Computational Linguistics.

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., and Le, Q. V. (2019). Xlnet: Generalized autoregressive pretraining for language understanding. *In Advances in neural information processing systems*, page 5753–5763.

Yang, Z., Salakhutdinov, R., and Cohen, W. W. (2016a). Multi-task cross-lingual sequence tagging from scratch. *CoRR*, abs/1603.06270, 2016.

Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., and Hovy, E. (2016b). Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489, San Diego, California. Association for Computational Linguistics.

Yates, A., Banko, M., Broadhead, M., Cafarella, M. J., Etzioni, O., and Soderland, S. (2007). Textrunner: open information extraction on the web. In *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*, pages 25–26.

Yates, A. and Etzioni, O. (2009). Unsupervised methods for determining object and relation synonyms on the web. *J. Artif. Int. Res.*, 34(1):255–296.

Yin, W., Kann, K., Yu, M., and Schütze, H. (2017). Comparative study of cnn and rnn for natural language processing. *arXiv preprint arXiv:1702.01923*.

Yoo, J. Y. and Qi, Y. (2021). Towards improving adversarial training of NLP models. *EMNLP*, abs/2109.00544.

Young, T., Hazarika, D., Poria, S., and Cambria, E. (2018). Recent trends in deep learning based natural language processing [review article]. *IEEE Computational Intelligence Magazine*, 13:55–75.

Yu, D., Wang, S., and Deng, L. (2010). Sequential labeling using deep-structured conditional random fields. *IEEE Journal of Selected Topics in Signal Processing*.

Yu, J., Qiu, M., Jiang, J., Huang, J., Song, S., Chu, W., and Chen, H. (2018). Modelling domain relationships for transfer learning on retrieval-based question answering systems in e-commerce. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, WSDM '18, page 682–690, New York, NY, USA. Association for Computing Machinery.

Zamparelli, R., Chowdhury, S., Brunato, D., Chesi, C., Dell'Orletta, F., Hasan, M. A., and Venturi, G. (2022). SemEval-2022 task 3: PreTENS-evaluating neural networks on presuppositional semantic knowledge. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 228–238, Seattle, United States. Association for Computational Linguistics.

Zeng, D., Liu, K., Chen, Y., and Zhao, J. (2015). Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1753–1762, Lisbon, Portugal. Association for Computational Linguistics.

Zhan, J. and Zhao, H. (2020). Span model for open information extraction on accurate corpus. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9523–9530.

Zhang, S. and Elhadad, N. (2013). Unsupervised biomedical named entity recognition: Experiments with clinical and biological texts. *Journal of biomedical informatics*, 46(6):1088–1098.

Zhang, T., Wu, F., Katiyar, A., Weinberger, K. Q., and Artzi, Y. (2021). Revisiting few-sample {bert} fine-tuning. In *International Conference on Learning Representations*.

Zhang, Y., Lin, J., Fan, Y., Jin, P., Liu, Y., and Liu, B. (2020). Cn-hit-it. nlp at semeval-2020 task 4: Enhanced language representation with multiple knowledge triples. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 494–500.

Zhao, Q., Tao, S., Zhou, J., Wang, L., Lin, X., and He, L. (2020). Ecnu-sensemaker at semeval-2020 task 4: Leveraging heterogeneous knowledge resources for commonsense validation and explanation. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 401–410, Barcelona (online). International Committee for Computational Linguistics.

Zhila, A. and Gelbukh, A. (2013). Comparison of open information extraction for english and spanish. *Computational Linguistics and Intelligent Technologies*, 12(19):714–722.

Zhong, H., Zhang, J., Wang, Z., Wan, H., and Chen, Z. (2015). Aligning knowledge and text embeddings by entity descriptions. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 267–272.

Zhou, G. and Su, J. (2002). Named entity recognition using an HMM-based chunk tagger. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 473–480.

Zhou, G., Su, J., Zhang, J., and Zhang, M. (2005). Exploring various knowledge in relation extraction. In *Proceedings of the 43rd annual meeting of the association for computational linguistics (acl'05)*, pages 427–434.

Zhu, K., San Wong, Y., and Hong, G. S. (2009). Multi-category micro-milling tool wear monitoring with continuous hidden markov models. *Mechanical Systems and Signal Processing*, 23(2):547–560.

Zhu, Q., Li, X., Conesa, A., and Pereira, C. (2018). GRAM-CNN: a deep learning approach with local context for named entity recognition in biomedical text. *Bioinformatics*, 34(9):1547–1554.

List of publications

1. Sarhan, I., Mosteiro, P., and Spruit, M. (2022). UU-tax at SemEval-2022 task 3: Improving the generalizability of language models for taxonomy classification through data augmentation. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 271–281, Seattle, United States. Association for Computational Linguistics

2. Sarhan, I. and Spruit, M. (2021). Open-cykg: An open cyber threat intelligence knowledge graph. *Knowledge-Based Systems*, 233:107524

3. van Haastrecht, M., Sarhan, I., Shojaifar, A., Baumgartner, L., Mallouli, W., and Spruit, M. (2021a). A threat-based cybersecurity risk assessment approach addressing sme needs. In *The 16th International Conference on Availability, Reliability and Security*, pages 1–12

4. van Haastrecht, M., Sarhan, I., Yigit Ozkan, B., Brinkhuis, M., and Spruit, M. (2021b). Symbals: A systematic review methodology blending active learning and snowballing. *Frontiers in research metrics and analytics*, 6:33

5. Sarhan, I. and Spruit, M. (2020). Can we survive without labelled data in NLP? Transfer learning for open information extraction. *Applied Sciences*, 10(17):5758

6. Sarhan, I. and Spruit, M. R. (2019). Contextualized word embeddings in a neural open information extraction model. In *Natural Language Processing and Information Systems*, pages 359–367, Cham. Springer International Publishing

7. Sarhan, I. and Spruit, M. (2018). Uncovering algorithmic approaches in open information extraction: A literature review. In *30th Benelux Conference on Artificial Intelligence*, pages 223–234. Springer CSAI/JADS

8. Sarhan, I., El-Sonbaty, Y., and Abou El-Nasr, M. (2016a). Semi-supervised pattern based algorithm for arabic relation extraction. In *2016 IEEE 28th international conference on tools with artificial intelligence (ICTAI)*, pages 177–183. IEEE

9. Sarhan, I., El-Sonbaty, Y., and El-Nasr, M. (2016b). Arabic relation extraction: a survey. *International Journal of Computer and Information Technology*, 5(5)

Summary

Natural Language Processing (NLP) develops computational techniques that permit automatic and expressive analysis of textual information within human language. The exponential growth of unstructured textual data has empowered the emergence of NLP-based techniques to extract valuable insights and information from this data. These can be used for various purposes such as sentiment analysis, content recommendation, question answering and more. NLP holds the promise of acquiring an unprecedented amount of knowledge from this textual data. A core NLP task that aids in the aforementioned applications is *Information Extraction*.

However, at present, existing information extraction systems lack two critical components. First, a methodology that is capable of extracting valuable information and which does not demand either a pre-defined set of relations or an existing ontology. This limits the extraction to a specific set of information and increases the probability of missing vital knowledge. Second, a proper data structure that supports storing the extracted knowledge in an efficient and effective way to enable more successful information retrieval and deeper knowledge understanding. Therefore, the main research question in this Ph.D. thesis is:

To what extent can we enhance open information extraction methods to efficiently and effectively represent unstructured textual data?

This research answers this question by designing an *Open Information Extraction (OIE)* framework that can be used as the primary building block to populate knowledge graphs, which results in an efficient textual data representation and facilitates querying by users.

In the first three research chapters of this work we focus primarily on OIE systems, along with the challenges associated with the design choices. Therefore, in Chapter 2, we start by exploring existing OIE methods to better comprehend the underlying limitations of the previously proposed approaches. More precisely, we analyze the performance of pattern-based methods alongside approaches that rely on machine learning classifiers to perform triple extraction. Additionally, we investigate the emerging trend of employing neural

techniques.

After the thorough analysis made in Chapter 2, we propose a Bidirectional Gated Recurrent Unit (Bi-GRU) OIE model that relies on contextualized word embeddings to extract relevant triples from unstructured text in Chapter 3, and we demonstrate the effectiveness of employing contextual embeddings in the OIE process by conducting various experiments. The proposed OIE supervised model, which addresses the task as a sequence tagging problem and employs a Bi-GRU, has achieved state-of-the-art performance and has been demonstrated to be effective in generating high-quality relation triples.

As the lack of labeled data is a long-standing problem that hinders the development of various NLP tasks, in Chapter 4, we extend the research done in Chapter 3 by relying upon learned features to generate relation triples to measure the transferability of our OIE model. We study how transferable these features are from one OIE domain to another. Further, we analyze their transferability to the semantically related NLP task of Relation Extraction (RE). We thereby contribute to answering the question whether *OIE can help achieve adequate NLP performance without labeled data*. With experiments carried out using both inductive transfer learning and transductive transfer learning, the results show comparable performance with traditional training on the downstream tasks.

In the second part of this dissertation we explore several procedures to *enhance the robustness of pre-trained language models* for the task of taxonomy classification in Chapter 5. Pre-trained language models often fail to generalize to unseen patterns during inference on downstream tasks. Given that a critical challenge regarding this task is the limited size of annotated data, we aim to detect the taxonomic relationship between nouns where the pattern or the set of nouns in the sentence is previously unseen. We propose a two-stage fine-tuning procedure using data augmentation techniques to improve the generalizability of pre-trained language models. In addition, rigorous experiments are carried out using multi-task learning and data-enriched fine-tuning on different state-of-the-art pre-trained language models. Our proposed two-stage fine-tuning model demonstrated strong generalizability on unseen data during inference, our proposed model had an F1 score of 91.25%. Moreover, to perform a regression task, we design a simple yet efficient model that relies on features generated from sentence encoders to train a classifier. In the regression sub-task, the simplicity of our model was a key factor in its success, resulting in competitive performance by achieving a ρ-score of 0.221.

Finally, in Chapter 6, we explore how OIE can be employed to construct a knowledge graph. Henceforth, we design and evaluate *Open-CyKG*: an open cyber threat intelligence knowledge graph framework that is constructed using

an attention-based neural OIE model to extract vital cyber threat information from unstructured advanced persistent threat reports. We further design a Named Entity Recognition (NER) model for automatically labeling cybersecurity terms. The knowledge graph is constructed using triples generated from the OIE model and with the aid of the NER model. One of the main complications during knowledge graph construction and population is ambiguity and data redundancy. To overcome this challenge, we conduct refinement and canonicalization techniques to fuse information in the knowledge graph based on contextualized word embeddings using hierarchical agglomerative clustering for entity fusion. In the final stage, we show that once Open-CyKG is created, querying can be performed efficiently, signifying that OIE can highly support the development of knowledge graphs. The attention-based OIE component, along with the CRF-NER model and knowledge graph canonicalization that constitute Open-CyKG, have all achieved beyond-state-of-the-art results, with the OIE component achieving an F-measure of 59.4% and the NER component scoring an F-measure of 98.9% on the Microsoft Security Bulletin dataset. Furthermore, the Knowledge graph canonicalization achieves a Macro F-measure of 82.6%, firmly establishing the effectiveness of the proposed methodology for representing knowledge.

Knowledge Graphs are architectures that, once constructed, can be used to imitate human intelligence that is useful for complex downstream applications such as recommender systems, search engines, dialog systems, and many more. The research and results presented in the previous chapters demonstrate that it is possible to significantly improve the efficiency and effectiveness of open information extraction methods for representing unstructured textual data through the use of techniques such as data augmentation, multi-stage fine-tuning, and pre-trained language models.

Samenvatting

Natural Language Processing (NLP) ontwikkelt computationele technieken die automatische en expressieve analyse van tekstuele informatie in menselijke taal mogelijk maken. De exponentiële groei van ongestructureerde tekstuele gegevens heeft de opkomst van op NLP gebaseerde technieken mogelijk gemaakt om waardevolle inzichten en informatie uit deze gegevens te halen. Deze kunnen worden gebruikt voor diverse doeleinden, zoals sentimentanalyse, aanbevelingen voor inhoud, beantwoording van vragen en meer. NLP houdt de belofte in van het verwerven van een ongekende hoeveelheid kennis uit deze tekstuele gegevens. Een kerntaak van NLP die helpt bij de bovengenoemde toepassingen is *Informatie-extractie*.

Momenteel ontbreekt het de bestaande informatie-extractiesystemen echter aan twee kritische componenten. Ten eerste, een methodologie die in staat is waardevolle informatie te extraheren en die geen vooraf gedefinieerde reeks relaties of een bestaande ontologie vereist. Dit beperkt de extractie tot een specifieke set van informatie en verhoogt de kans op het missen van vitale kennis. Ten tweede, een goede gegevensstructuur die het opslaan van de geëxtraheerde kennis op een efficiënte en effectieve manier ondersteunt, zodat informatie succesvoller kan worden teruggevonden en kennis beter kan worden begrepen. Daarom is de centrale onderzoeksvraag in dit proefschrift:

In hoeverre kunnen we open informatie-extractiemethoden verbeteren om ongestructureerde tekstuele gegevens efficiënt en effectief te representeren?

Dit onderzoek beantwoordt deze vraag door een OIE-raamwerk (Open Informatie-extractie) te ontwerpen dat kan worden gebruikt als de primaire bouwsteen om kennisgrafen te vullen, wat resulteert in een efficiënte tekstuele gegevensrepresentatie en wat het bevragen door gebruikers vergemakkelijkt.

In de eerste drie onderzoekshoofdstukken van dit werk richten we ons voornamelijk op OIE-systemen, samen met de uitdagingen die gepaard gaan met de ontwerpkeuzes. Daarom beginnen we in Hoofdstuk 2 met het verkennen van bestaande OIE-methoden om de onderliggende beperkingen van de eerder voorgestelde benaderingen beter te begrijpen. In het bijzonder analyseren we

de prestaties van patroongebaseerde methoden naast benaderingen die vertrouwen op machine-geleerde patroonherkenners om triplet-extractie uit te voeren. Bovendien onderzoeken we de opkomende trend om neurale technieken in te zetten.

Na de grondige analyse gemaakt in Hoofdstuk 2, stellen we in Hoofdstuk 3 een Bidirectional Gated Recurrent Unit (Bi-GRU) OIE model voor dat vertrouwt op gecontextualiseerde woord-inbeddingen om relevante tripletten te extraheren uit ongestructureerde tekst, en tonen we de effectiviteit van het gebruik van contextuele inbeddingen in het OIE proces aan door het uitvoeren van verschillende experimenten. Het voorgestelde OIE-model, dat de taak adresseert als een sequentiemarkeringsprobleem en dat gebruik maakt van een Bi-GRU, heeft state-of-the-art prestaties behaald en is effectief gebleken in het genereren van hoogwaardige relatietripletten.

Omdat het gebrek aan gelabelde gegevens een al lang bestaand probleem is dat de ontwikkeling van verschillende NLP-taken belemmert, breiden we in Hoofdstuk 4 het onderzoek in hoofdstuk 3 uit door te vertrouwen op geleerde kenmerken om relatietripletten te genereren teneinde de overdraagbaarheid van ons OIE-model te meten. We bestuderen hoe overdraagbaar deze kenmerken zijn van het ene OIE-domein naar het andere. Verder analyseren we hun overdraagbaarheid op de semantisch verwante NLP taak van Relatie-extractie (RE). Zo dragen we bij tot de beantwoording van de vraag of OIE kan helpen om adequate NLP-prestaties te bereiken zonder gelabelde gegevens. Experimenten uitgevoerd met zowel inductief overdrachtsleren als transductief overdrachtsleren, tonen vergelijkbare prestaties met traditionele training op afgeleide NLP-taken.

In het tweede deel van dit proefschrift onderzoeken we verschillende procedures om de robuustheid van voorgetrainde taalmodellen te verbeteren voor de taak van taxonomieclassificatie in Hoofdstuk 5. Voorgetrainde taalmodellen generaliseren vaak niet naar ongeziene patronen tijdens inferentie op downstream taken. Aangezien de beperkte omvang van geannoteerde gegevens een kritieke uitdaging vormt voor deze taak, streven wij ernaar de taxonomische relatie tussen zelfstandige naamwoorden te detecteren wanneer het patroon of de verzameling zelfstandige naamwoorden in de zin nog niet eerder is gezien. Wij stellen een tweefasige fijnafstemmingsprocedure voor met behulp van data-uitbreidingstechnieken om de generaliseerbaarheid van voorgetrainde taalmodellen te verbeteren. Bovendien worden grondige experimenten uitgevoerd met multitaak leren en data-verrijkte fijnafstemming op verschillende state-of-the-art voorgetrainde taalmodellen. Ons voorgestelde tweetraps fijnafstemmingsmodel toonde een sterke generaliseerbaarheid op ongeziene data tijdens inferentie, ons voorgestelde model had een F1-score van 91,25%. Om

een regressietaak uit te voeren, ontwerpen we bovendien een eenvoudig maar efficiënt model dat zich baseert op kenmerken gegenereerd door zincoders om een patroonherkenner te trainen. In de subtaak regressie was de eenvoud van ons model een belangrijke factor voor het succes ervan, resulterend in competitieve prestaties met een ρ-score van 0,221.

Ten slotte onderzoeken we in Hoofdstuk 6 hoe OIE kan worden gebruikt om een kennisgraaf te construeren. Vervolgens ontwerpen en evalueren we *Open-CyKG*: een open cyberdreiging inlichtingen kennisgraaf raamwerk dat is opgebouwd met behulp van een op aandacht gebaseerd neuraal OIE-model om vitale cyberdreigingsinformatie te extraheren uit ongestructureerde rapporten over geavanceerde aanhoudende dreigingen. Verder ontwerpen we een Entiteit-herkennings(NER)-model voor het automatisch labelen van cyberdreigingstermen. De kennisgraaf wordt opgebouwd met behulp van tripletten gegenereerd uit het OIE-model en met behulp van het NER-model. Een van de belangrijkste complicaties bij de constructie en populatie van de kennisgraaf is ambiguïteit en redundantie van gegevens. Wij gaan deze uitdaging aan door verfijnings- en canonicaliseringstechnieken toe te passen om informatie in de kennisgraaf samen te voegen op basis van gecontextualiseerde woordinbeddingen met behulp van hiërarchische agglomeratieve clustering voor entiteitsfusie. In de laatste fase laten we zien dat zodra Open-CyKG is gecreëerd, kennisvragen efficiënt kunnen worden uitgevoerd, wat aangeeft dat OIE de ontwikkeling van kennisgrafen in hoge mate kan ondersteunen. De op aandacht gebaseerde OIE-component, het CRF-NER-model en de kennisgraafcanonicalisatie, die tezamen Open-CyKG vormen, hebben allen beter dan state-of-the-art resultaten opgeleverd, waarbij de OIE-component een F-waarde van 59,4% en de NER-component een F-waarde van 98,9% behaalde op de Microsoft Security Bulletin-dataset. Bovendien behaalt de canonicalisatie van de kennisgraaf een Macro F-waarde van 82,6%, waarmee de doeltreffendheid van de hier voorgestelde methodologie voor de representatie van kennis duidelijk wordt aangetoond.

Kennisgrafen zijn architecturen die, eenmaal geconstrueerd, kunnen worden gebruikt om menselijke intelligentie te imiteren, wat nuttig is voor complexe afgeleide toepassingen zoals aanbevelingssystemen, zoekmachines, dialoogsystemen, en nog veel meer. Het onderzoek en de resultaten in de voorgaande hoofdstukken tonen aan dat het mogelijk is om de efficiëntie en effectiviteit van open informatie-extractiemethoden voor de representatie van ongestructureerde tekstuele gegevens aanzienlijk te verbeteren door het gebruik van technieken zoals gegevensuitbreiding, meerfasige fijnafstemming en voorgetrainde taalmodellen.

Curriculum Vitae

Injy Sarhan was born on February 24th, 1992, in Alexandria, Egypt. She attended the Arab Academy For Science, Technology and Maritime Transport, where she obtained a Bachelor's and a Master's degrees in Computer Engineering in 2011 and 2014, respectively. She wrote her Master's thesis on *Arabic Relation Extraction*, where she designed an evaluated a semi-supervised pattern-based algorithm for extracting relations from unstructured textual data.

In 2018, she started her research as a part-time Ph.D. candidate at the department of Information and Computing Sciences of Utrecht University. Injy worked remotely from Egypt and scheduled regular video calls with her supervisor to discuss research progress. She also visited Utrecht University twice per year for a period of a month each time. During her visits, Injy joins the department's research meetings and the ADS lab colloquia. Simultaneous with doing her research, Injy has been acting as a teaching assistant at the department of Computer Engineering at her home university. Among her teaching duties were preparing and conducting tutorial and lab sessions for various bachelor courses, including Computer Networks, Operating Systems, Programming Applications, and Pattern Recognition. Additionally, she was a guest lecturer at the University of Bremen, Germany, for the summer courses in years 2017 and 2019, teaching Database Design using SQL and Linux Command Line Interface. In the course of her time as a Ph.D. researcher, She presented her work at various international scientific meetings, including the SemEval workshop, co-located with the North American Association for Computational (NAACL).

In October 2020, Injy moved to the Netherlands to continue her Ph.D. degree and joined the Horizon2020 project, CyberGEIGER[2], in collaboration with several European partners where she contributed to designing GEIGER Risk Indicator that aims to help small-sized enterprises to manage their cybersecurity risks through recommending countermeasures to mitigate those risks.

[2]https://project.cyber-geiger.eu/

SIKS Dissertation Series

2016 01 Syed Saiden Abbas (RUN), Recognition of Shapes by Humans and Machines
02 Michiel Christiaan Meulendijk (UU), Optimizing medication reviews through decision support: prescribing a better pill to swallow
03 Maya Sappelli (RUN), Knowledge Work in Context: User Centered Knowledge Worker Support
04 Laurens Rietveld (VU), Publishing and Consuming Linked Data
05 Evgeny Sherkhonov (UVA), Expanded Acyclic Queries: Containment and an Application in Explaining Missing Answers
06 Michel Wilson (TUD), Robust scheduling in an uncertain environment
07 Jeroen de Man (VU), Measuring and modeling negative emotions for virtual training
08 Matje van de Camp (TiU), A Link to the Past: Constructing Historical Social Networks from Unstructured Data
09 Archana Nottamkandath (VU), Trusting Crowdsourced Information on Cultural Artefacts
10 George Karafotias (VUA), Parameter Control for Evolutionary Algorithms
11 Anne Schuth (UVA), Search Engines that Learn from Their Users
12 Max Knobbout (UU), Logics for Modelling and Verifying Normative Multi-Agent Systems
13 Nana Baah Gyan (VU), The Web, Speech Technologies and Rural Development in West Africa - An ICT4D Approach
14 Ravi Khadka (UU), Revisiting Legacy Software System Modernization
15 Steffen Michels (RUN), Hybrid Probabilistic Logics - Theoretical Aspects, Algorithms and Experiments
16 Guangliang Li (UVA), Socially Intelligent Autonomous Agents that Learn from Human Reward
17 Berend Weel (VU), Towards Embodied Evolution of Robot Organisms

18 Albert Meroño Peñuela (VU), Refining Statistical Data on the Web
19 Julia Efremova (Tu/e), Mining Social Structures from Genealogical Data
20 Daan Odijk (UVA), Context & Semantics in News & Web Search
21 Alejandro Moreno Célleri (UT), From Traditional to Interactive Playspaces: Automatic Analysis of Player Behavior in the Interactive Tag Playground
22 Grace Lewis (VU), Software Architecture Strategies for Cyber-Foraging Systems
23 Fei Cai (UVA), Query Auto Completion in Information Retrieval
24 Brend Wanders (UT), Repurposing and Probabilistic Integration of Data; An Iterative and data model independent approach
25 Julia Kiseleva (TU/e), Using Contextual Information to Understand Searching and Browsing Behavior
26 Dilhan Thilakarathne (VU), In or Out of Control: Exploring Computational Models to Study the Role of Human Awareness and Control in Behavioural Choices, with Applications in Aviation and Energy Management Domains
27 Wen Li (TUD), Understanding Geo-spatial Information on Social Media
28 Mingxin Zhang (TUD), Large-scale Agent-based Social Simulation - A study on epidemic prediction and control
29 Nicolas Höning (TUD), Peak reduction in decentralised electricity systems - Markets and prices for flexible planning
30 Ruud Mattheij (UvT), The Eyes Have It
31 Mohammad Khelghati (UT), Deep web content monitoring
32 Eelco Vriezekolk (UT), Assessing Telecommunication Service Availability Risks for Crisis Organisations
33 Peter Bloem (UVA), Single Sample Statistics, exercises in learning from just one example
34 Dennis Schunselaar (TUE), Configurable Process Trees: Elicitation, Analysis, and Enactment
35 Zhaochun Ren (UVA), Monitoring Social Media: Summarization, Classification and Recommendation
36 Daphne Karreman (UT), Beyond R2D2: The design of nonverbal interaction behavior optimized for robot-specific morphologies
37 Giovanni Sileno (UvA), Aligning Law and Action - a conceptual and computational inquiry
38 Andrea Minuto (UT), Materials that Matter - Smart Materials meet Art & Interaction Design

39 Merijn Bruijnes (UT), Believable Suspect Agents; Response and Interpersonal Style Selection for an Artificial Suspect
40 Christian Detweiler (TUD), Accounting for Values in Design
41 Thomas King (TUD), Governing Governance: A Formal Framework for Analysing Institutional Design and Enactment Governance
42 Spyros Martzoukos (UVA), Combinatorial and Compositional Aspects of Bilingual Aligned Corpora
43 Saskia Koldijk (RUN), Context-Aware Support for Stress Self-Management: From Theory to Practice
44 Thibault Sellam (UVA), Automatic Assistants for Database Exploration
45 Bram van de Laar (UT), Experiencing Brain-Computer Interface Control
46 Jorge Gallego Perez (UT), Robots to Make you Happy
47 Christina Weber (UL), Real-time foresight - Preparedness for dynamic innovation networks
48 Tanja Buttler (TUD), Collecting Lessons Learned
49 Gleb Polevoy (TUD), Participation and Interaction in Projects. A Game-Theoretic Analysis
50 Yan Wang (UVT), The Bridge of Dreams: Towards a Method for Operational Performance Alignment in IT-enabled Service Supply Chains

---

2017 01 Jan-Jaap Oerlemans (UL), Investigating Cybercrime
02 Sjoerd Timmer (UU), Designing and Understanding Forensic Bayesian Networks using Argumentation
03 Daniël Harold Telgen (UU), Grid Manufacturing; A Cyber-Physical Approach with Autonomous Products and Reconfigurable Manufacturing Machines
04 Mrunal Gawade (CWI), Multi-core Parallelism in a Column-store
05 Mahdieh Shadi (UVA), Collaboration Behavior
06 Damir Vandic (EUR), Intelligent Information Systems for Web Product Search
07 Roel Bertens (UU), Insight in Information: from Abstract to Anomaly
08 Rob Konijn (VU) , Detecting Interesting Differences:Data Mining in Health Insurance Data using Outlier Detection and Subgroup Discovery
09 Dong Nguyen (UT), Text as Social and Cultural Data: A Computational Perspective on Variation in Text

10 Robby van Delden (UT), (Steering) Interactive Play Behavior
11 Florian Kunneman (RUN), Modelling patterns of time and emotion in Twitter #anticipointment
12 Sander Leemans (TUE), Robust Process Mining with Guarantees
13 Gijs Huisman (UT), Social Touch Technology - Extending the reach of social touch through haptic technology
14 Shoshannah Tekofsky (UvT), You Are Who You Play You Are: Modelling Player Traits from Video Game Behavior
15 Peter Berck (RUN), Memory-Based Text Correction
16 Aleksandr Chuklin (UVA), Understanding and Modeling Users of Modern Search Engines
17 Daniel Dimov (UL), Crowdsourced Online Dispute Resolution
18 Ridho Reinanda (UVA), Entity Associations for Search
19 Jeroen Vuurens (UT), Proximity of Terms, Texts and Semantic Vectors in Information Retrieval
20 Mohammadbashir Sedighi (TUD), Fostering Engagement in Knowledge Sharing: The Role of Perceived Benefits, Costs and Visibility
21 Jeroen Linssen (UT), Meta Matters in Interactive Storytelling and Serious Gaming (A Play on Worlds)
22 Sara Magliacane (VU), Logics for causal inference under uncertainty
23 David Graus (UVA), Entities of Interest — Discovery in Digital Traces
24 Chang Wang (TUD), Use of Affordances for Efficient Robot Learning
25 Veruska Zamborlini (VU), Knowledge Representation for Clinical Guidelines, with applications to Multimorbidity Analysis and Literature Search
26 Merel Jung (UT), Socially intelligent robots that understand and respond to human touch
27 Michiel Joosse (UT), Investigating Positioning and Gaze Behaviors of Social Robots: People's Preferences, Perceptions and Behaviors
28 John Klein (VU), Architecture Practices for Complex Contexts
29 Adel Alhuraibi (UvT), From IT-BusinessStrategic Alignment to Performance: A Moderated Mediation Model of Social Innovation, and Enterprise Governance of IT"
30 Wilma Latuny (UvT), The Power of Facial Expressions
31 Ben Ruijl (UL), Advances in computational methods for QFT calculations
32 Thaer Samar (RUN), Access to and Retrievability of Content in Web Archives

33 Brigit van Loggem (OU), Towards a Design Rationale for Software Documentation: A Model of Computer-Mediated Activity
34 Maren Scheffel (OU), The Evaluation Framework for Learning Analytics
35 Martine de Vos (VU), Interpreting natural science spreadsheets
36 Yuanhao Guo (UL), Shape Analysis for Phenotype Characterisation from High-throughput Imaging
37 Alejandro Montes Garcia (TUE), WiBAF: A Within Browser Adaptation Framework that Enables Control over Privacy
38 Alex Kayal (TUD), Normative Social Applications
39 Sara Ahmadi (RUN), Exploiting properties of the human auditory system and compressive sensing methods to increase noise robustness in ASR
40 Altaf Hussain Abro (VUA), Steer your Mind: Computational Exploration of Human Control in Relation to Emotions, Desires and Social Support For applications in human-aware support systems
41 Adnan Manzoor (VUA), Minding a Healthy Lifestyle: An Exploration of Mental Processes and a Smart Environment to Provide Support for a Healthy Lifestyle
42 Elena Sokolova (RUN), Causal discovery from mixed and missing data with applications on ADHD datasets
43 Maaike de Boer (RUN), Semantic Mapping in Video Retrieval
44 Garm Lucassen (UU), Understanding User Stories - Computational Linguistics in Agile Requirements Engineering
45 Bas Testerink (UU), Decentralized Runtime Norm Enforcement
46 Jan Schneider (OU), Sensor-based Learning Support
47 Jie Yang (TUD), Crowd Knowledge Creation Acceleration
48 Angel Suarez (OU), Collaborative inquiry-based learning

2018 01 Han van der Aa (VUA), Comparing and Aligning Process Representations
02 Felix Mannhardt (TUE), Multi-perspective Process Mining
03 Steven Bosems (UT), Causal Models For Well-Being: Knowledge Modeling, Model-Driven Development of Context-Aware Applications, and Behavior Prediction
04 Jordan Janeiro (TUD), Flexible Coordination Support for Diagnosis Teams in Data-Centric Engineering Tasks
05 Hugo Huurdeman (UVA), Supporting the Complex Dynamics of the Information Seeking Process

06 Dan Ionita (UT), Model-Driven Information Security Risk Assessment of Socio-Technical Systems
07 Jieting Luo (UU), A formal account of opportunism in multi-agent systems
08 Rick Smetsers (RUN), Advances in Model Learning for Software Systems
09 Xu Xie (TUD), Data Assimilation in Discrete Event Simulations
10 Julienka Mollee (VUA), Moving forward: supporting physical activity behavior change through intelligent technology
11 Mahdi Sargolzaei (UVA), Enabling Framework for Service-oriented Collaborative Networks
12 Xixi Lu (TUE), Using behavioral context in process mining
13 Seyed Amin Tabatabaei (VUA), Computing a Sustainable Future
14 Bart Joosten (UVT), Detecting Social Signals with Spatiotemporal Gabor Filters
15 Naser Davarzani (UM), Biomarker discovery in heart failure
16 Jaebok Kim (UT), Automatic recognition of engagement and emotion in a group of children
17 Jianpeng Zhang (TUE), On Graph Sample Clustering
18 Henriette Nakad (UL), De Notaris en Private Rechtspraak
19 Minh Duc Pham (VUA), Emergent relational schemas for RDF
20 Manxia Liu (RUN), Time and Bayesian Networks
21 Aad Slootmaker (OUN), EMERGO: a generic platform for authoring and playing scenario-based serious games
22 Eric Fernandes de Mello Araújo (VUA), Contagious: Modeling the Spread of Behaviours, Perceptions and Emotions in Social Networks
23 Kim Schouten (EUR), Semantics-driven Aspect-Based Sentiment Analysis
24 Jered Vroon (UT), Responsive Social Positioning Behaviour for Semi-Autonomous Telepresence Robots
25 Riste Gligorov (VUA), Serious Games in Audio-Visual Collections
26 Roelof Anne Jelle de Vries (UT),Theory-Based and Tailor-Made: Motivational Messages for Behavior Change Technology
27 Maikel Leemans (TUE), Hierarchical Process Mining for Scalable Software Analysis
28 Christian Willemse (UT), Social Touch Technologies: How they feel and how they make you feel
29 Yu Gu (UVT), Emotion Recognition from Mandarin Speech

	30	Wouter Beek, The "K" in "semantic web" stands for "knowledge": scaling semantics to the web
2019	01	Rob van Eijk (UL),Web privacy measurement in real-time bidding systems. A graph-based approach to RTB system classification
	02	Emmanuelle Beauxis Aussalet (CWI, UU), Statistics and Visualizations for Assessing Class Size Uncertainty
	03	Eduardo Gonzalez Lopez de Murillas (TUE), Process Mining on Databases: Extracting Event Data from Real Life Data Sources
	04	Ridho Rahmadi (RUN), Finding stable causal structures from clinical data
	05	Sebastiaan van Zelst (TUE), Process Mining with Streaming Data
	06	Chris Dijkshoorn (VU), Nichesourcing for Improving Access to Linked Cultural Heritage Datasets
	07	Soude Fazeli (TUD), Recommender Systems in Social Learning Platforms
	08	Frits de Nijs (TUD), Resource-constrained Multi-agent Markov Decision Processes
	09	Fahimeh Alizadeh Moghaddam (UVA), Self-adaptation for energy efficiency in software systems
	10	Qing Chuan Ye (EUR), Multi-objective Optimization Methods for Allocation and Prediction
	11	Yue Zhao (TUD), Learning Analytics Technology to Understand Learner Behavioral Engagement in MOOCs
	12	Jacqueline Heinerman (VU), Better Together
	13	Guanliang Chen (TUD), MOOC Analytics: Learner Modeling and Content Generation
	14	Daniel Davis (TUD), Large-Scale Learning Analytics: Modeling Learner Behavior & Improving Learning Outcomes in Massive Open Online Courses
	15	Erwin Walraven (TUD), Planning under Uncertainty in Constrained and Partially Observable Environments
	16	Guangming Li (TUE), Process Mining based on Object-Centric Behavioral Constraint (OCBC) Models
	17	Ali Hurriyetoglu (RUN),Extracting actionable information from microtexts
	18	Gerard Wagenaar (UU), Artefacts in Agile Team Communication
	19	Vincent Koeman (TUD), Tools for Developing Cognitive Agents
	20	Chide Groenouwe (UU), Fostering technically augmented human collective intelligence

21 Cong Liu (TUE), Software Data Analytics: Architectural Model Discovery and Design Pattern Detection
22 Martin van den Berg (VU),Improving IT Decisions with Enterprise Architecture
23 Qin Liu (TUD), Intelligent Control Systems: Learning, Interpreting, Verification
24 Anca Dumitrache (VU), Truth in Disagreement - Crowdsourcing Labeled Data for Natural Language Processing
25 Emiel van Miltenburg (VU), Pragmatic factors in (automatic) image description
26 Prince Singh (UT), An Integration Platform for Synchromodal Transport
27 Alessandra Antonaci (OUN), The Gamification Design Process applied to (Massive) Open Online Courses
28 Esther Kuindersma (UL), Cleared for take-off: Game-based learning to prepare airline pilots for critical situations
29 Daniel Formolo (VU), Using virtual agents for simulation and training of social skills in safety-critical circumstances
30 Vahid Yazdanpanah (UT), Multiagent Industrial Symbiosis Systems
31 Milan Jelisavcic (VU), Alive and Kicking: Baby Steps in Robotics
32 Chiara Sironi (UM), Monte-Carlo Tree Search for Artificial General Intelligence in Games
33 Anil Yaman (TUE), Evolution of Biologically Inspired Learning in Artificial Neural Networks
34 Negar Ahmadi (TUE), EEG Microstate and Functional Brain Network Features for Classification of Epilepsy and PNES
35 Lisa Facey-Shaw (OUN), Gamification with digital badges in learning programming
36 Kevin Ackermans (OUN), Designing Video-Enhanced Rubrics to Master Complex Skills
37 Jian Fang (TUD), Database Acceleration on FPGAs
38 Akos Kadar (OUN), Learning visually grounded and multilingual representations

---

2020 01 Armon Toubman (UL), Calculated Moves: Generating Air Combat Behaviour
02 Marcos de Paula Bueno (UL), Unraveling Temporal Processes using Probabilistic Graphical Models
03 Mostafa Deghani (UvA), Learning with Imperfect Supervision for Language Understanding

04 Maarten van Gompel (RUN), Context as Linguistic Bridges
05 Yulong Pei (TUE), On local and global structure mining
06 Preethu Rose Anish (UT), Stimulation Architectural Thinking during Requirements Elicitation - An Approach and Tool Support
07 Wim van der Vegt (OUN), Towards a software architecture for reusable game components
08 Ali Mirsoleimani (UL),Structured Parallel Programming for Monte Carlo Tree Search
09 Myriam Traub (UU), Measuring Tool Bias and Improving Data Quality for Digital Humanities Research
10 Alifah Syamsiyah (TUE), In-database Preprocessing for Process Mining
11 Sepideh Mesbah (TUD), Semantic-Enhanced Training Data AugmentationMethods for Long-Tail Entity Recognition Models
12 Ward van Breda (VU), Predictive Modeling in E-Mental Health: Exploring Applicability in Personalised Depression Treatment
13 Marco Virgolin (CWI), Design and Application of Gene-pool Optimal Mixing Evolutionary Algorithms for Genetic Programming
14 Mark Raasveldt (CWI/UL), Integrating Analytics with Relational Databases
15 Konstantinos Georgiadis (OUN), Smart CAT: Machine Learning for Configurable Assessments in Serious Games
16 Ilona Wilmont (RUN), Cognitive Aspects of Conceptual Modelling
17 Daniele Di Mitri (OUN), The Multimodal Tutor: Adaptive Feedback from Multimodal Experiences
18 Georgios Methenitis (TUD), Agent Interactions & Mechanisms in Markets with Uncertainties: Electricity Markets in Renewable Energy Systems
19 Guido van Capelleveen (UT), Industrial Symbiosis Recommender Systems
20 Albert Hankel (VU), Embedding Green ICT Maturity in Organisations
21 Karine da Silva Miras de Araujo (VU), Where is the robot?: Life as it could be
22 Maryam Masoud Khamis (RUN), Understanding complex systems implementation through a modeling approach: the case of e-government in Zanzibar
23 Rianne Conijn (UT), The Keys to Writing: A writing analytics approach to studying writing processes using keystroke logging

24 Lenin da Nóbrega Medeiros (VUA/RUN), How are you feeling, human? Towards emotionally supportive chatbots
25 Xin Du (TUE), The Uncertainty in Exceptional Model Mining
26 Krzysztof Leszek Sadowski (UU), GAMBIT: Genetic Algorithm for Model-Based mixed-Integer opTimization
27 Ekaterina Muravyeva (TUD), Personal data and informed consent in an educational context
28 Bibeg Limbu (TUD), Multimodal interaction for deliberate practice: Training complex skills with augmented reality
29 Ioan Gabriel Bucur (RUN), Being Bayesian about Causal Inference
30 Bob Zadok Blok (UL), Creatief, Creatieve, Creatiefst
31 Gongjin Lan (VU), Learning better – From Baby to Better
32 Jason Rhuggenaath (TUE), Revenue management in online markets: pricing and online advertising
33 Rick Gilsing (TUE), Supporting service-dominant business model evaluation in the context of business model innovation
34 Anna Bon (MU), Intervention or Collaboration? Redesigning Information and Communication Technologies for Development
35 Siamak Farshidi (UU), Multi-Criteria Decision-Making in Software Production

---

2021 01 Francisco Xavier Dos Santos Fonseca (TUD),Location-based Games for Social Interaction in Public Space
02 Rijk Mercuur (TUD), Simulating Human Routines: Integrating Social Practice Theory in Agent-Based Models
03 Seyyed Hadi Hashemi (UVA), Modeling Users Interacting with Smart Devices
04 Ioana Jivet (OU), The Dashboard That Loved Me: Designing adaptive learning analytics for self-regulated learning
05 Davide Dell'Anna (UU), Data-Driven Supervision of Autonomous Systems
06 Daniel Davison (UT), "Hey robot, what do you think?" How children learn with a social robot
07 Armel Lefebvre (UU), Research data management for open science
08 Nardie Fanchamps (OU), The Influence of Sense-Reason-Act Programming on Computational Thinking
09 Cristina Zaga (UT), The Design of Robothings. Non-Anthropomorphic and Non-Verbal Robots to Promote Children's Collaboration Through Play

10 Quinten Meertens (UvA), Misclassification Bias in Statistical Learning
11 Anne van Rossum (UL), Nonparametric Bayesian Methods in Robotic Vision
12 Lei Pi (UL), External Knowledge Absorption in Chinese SMEs
13 Bob R. Schadenberg (UT), Robots for Autistic Children: Understanding and Facilitating Predictability for Engagement in Learning
14 Negin Samaeemofrad (UL), Business Incubators: The Impact of Their Support
15 Onat Ege Adali (TU/e), Transformation of Value Propositions into Resource Re-Configurations through the Business Services Paradigm
16 Esam A. H. Ghaleb (UM), Bimodal emotion recognition from audio-visual cues
17 Dario Dotti (UM), Human Behavior Understanding from motion and bodily cues using deep neural networks
18 Remi Wieten (UU), Bridging the Gap Between Informal Sense-Making Tools and Formal Systems - Facilitating the Construction of Bayesian Networks and Argumentation Frameworks
19 Roberto Verdecchia (VU), Architectural Technical Debt: Identification and Management
20 Masoud Mansoury (TU/e), Understanding and Mitigating Multi-Sided Exposure Bias in Recommender Systems
21 Pedro Thiago Timbó Holanda (CWI), Progressive Indexes
22 Sihang Qiu (TUD), Conversational Crowdsourcing
23 Hugo Manuel Proença (LIACS), Robust rules for prediction and description
24 Kaijie Zhu (TUE), On Efficient Temporal Subgraph Query Processing
25 Eoin Martino Grua (VUA), The Future of E-Health is Mobile: Combining AI and Self-Adaptation to Create Adaptive E-Health Mobile Applications
26 Benno Kruit (CWI & VUA), Reading the Grid: Extending Knowledge Bases from Human-readable Tables
27 Jelte van Waterschoot (UT), Personalized and Personal Conversations: Designing Agents Who Want to Connect With You
28 Christoph Selig (UL), Understanding the Heterogeneity of Corporate Entrepreneurship Programs

2022 01 Judith van Stegeren (UT), Flavor text generation for role-playing video games
02 Paulo da Costa (TU/e), Data-driven Prognostics and Logistics Optimisation: A Deep Learning Journey
03 Ali el Hassouni (VUA), A Model A Day Keeps The Doctor Away: Reinforcement Learning For Personalized Healthcare
04 Ünal Aksu (UU), A Cross-Organizational Process Mining Framework
05 Shiwei Liu (TU/e), Sparse Neural Network Training with In-Time Over-Parameterization
06 Reza Refaei Afshar (TU/e), Machine Learning for Ad Publishers in Real Time Bidding
07 Sambit Praharaj (OU), Measuring the Unmeasurable? Towards Automatic Co-located Collaboration Analytics
08 Maikel L. van Eck (TU/e), Process Mining for Smart Product Design
09 Oana Andreea Inel (VUA), Understanding Events: A Diversity-driven Human-Machine Approach
10 Felipe Moraes Gomes (TUD), Examining the Effectiveness of Collaborative Search Engines
11 Mirjam de Haas (UT), Staying engaged in child-robot interaction, a quantitative approach to studying preschoolers' engagement with robots and tasks during second-language tutoring
12 Guanyi Chen (UU), Computational Generation of Chinese Noun Phrases
13 Xander Wilcke (VUA), Machine Learning on Multimodal Knowledge Graphs: Opportunities, Challenges, and Methods for Learning on Real-World Heterogeneous and Spatially-Oriented Knowledge
14 Michiel Overeem (UU), Evolution of Low-Code Platforms
15 Jelmer Jan Koorn (UU), Work in Process: Unearthing Meaning using Process Mining
16 Pieter Gijsbers (TU/e), Systems for AutoML Research
17 Laura van der Lubbe (VUA), Empowering vulnerable people with serious games and gamification
18 Paris Mavromoustakos Blom (TiU), Player Affect Modelling and Video Game Personalisation
19 Bilge Yigit Ozkan (UU), Cybersecurity Maturity Assessment and Standardisation
20 Fakhra Jabeen (VUA), Dark Side of the Digital Media - Computational Analysis of Negative Human Behaviors on Social Media

21 Seethu Mariyam Christopher (UM), Intelligent Toys for Physical and Cognitive Assessments
22 Alexandra Sierra Rativa (TiU), Virtual Character Design and its potential to foster Empathy, Immersion, and Collaboration Skills in Video Games and Virtual Reality Simulations
23 Ilir Kola (TUD), Enabling Social Situation Awareness in Support Agents
24 Samaneh Heidari (UU), Agents with Social Norms and Values - A framework for agent based social simulations with social norms and personal values
25 Anna L.D. Latour (LU), Optimal decision-making under constraints and uncertainty
26 Anne Dirkson (LU), Knowledge Discovery from Patient Forums: Gaining novel medical insights from patient experiences
27 Christos Athanasiadis (UM), Emotion-aware cross-modal domain adaptation in video sequences
28 Onuralp Ulusoy (UU), Privacy in Collaborative Systems
29 Jan Kolkmeier (UT), From Head Transform to Mind Transplant: Social Interactions in Mixed Reality
30 Dean De Leo (CWI), Analysis of Dynamic Graphs on Sparse Arrays
31 Konstantinos Traganos (TU/e), Tackling Complexity in Smart Manufacturing with Advanced Manufacturing Process Management
32 Cezara Pastrav (UU), Social simulation for socio-ecological systems
33 Brinn Hekkelman (CWI/TUD), Fair Mechanisms for Smart Grid Congestion Management
34 Nimat Ullah (VUA), Mind Your Behaviour: Computational Modelling of Emotion & Desire Regulation for Behaviour Change
35 Mike E.U. Ligthart (VUA), Shaping the Child-Robot Relationship: Interaction Design Patterns for a Sustainable Interaction

---

2023 01 Bojan Simoski (VUA), Untangling the Puzzle of Digital Health Interventions
02 Mariana Rachel Dias da Silva (TiU), Grounded or in flight? What our bodies can tell us about the whereabouts of our thoughts
03 Shabnam Najafian (TUD), User Modeling for Privacy-preserving Explanations in Group Recommendations
04 Gineke Wiggers (UL), The Relevance of Impact: bibliometric-enhanced legal information retrieval

05 Anton Bouter (CWI), Optimal Mixing Evolutionary Algorithms for Large-Scale Real-Valued Optimization, Including Real-World Medical Applications
06 António Pereira Barata (UL), Reliable and Fair Machine Learning for Risk Assessment
07 Tianjin Huang (TU/e), The Roles of Adversarial Examples on Trustworthiness of Deep Learning
08 Lu Yin (TU/e), Knowledge Elicitation using Psychometric Learning
09 Xu Wang (VUA), Scientific Dataset Recommendation with Semantic Techniques
10 Dennis J.N.J. Soemers (UM), Learning State-Action Features for General Game Playing
11 Fawad Taj (VUA), Towards Motivating Machines: Computational Modeling of the Mechanism of Actions for Effective Digital Health Behavior Change Applications
12 Tessel Bogaard (VUA), Using Metadata to Understand Search Behavior in Digital Libraries
13 Injy Sarhan (UU), Open Information Extraction for Knowledge Representation
14 Selma Čaušević (TUD), Energy resilience through self-organization
15 Alvaro Henrique Chaim Correia (TU/e), Insights on Learning Tractable Probabilistic Graphical Models
16 Peter Blomsma (TiU), Building Embodied Conversational Agents: Observations on human nonverbal behaviour as a resource for the development of artificial characters

To what extent can we enhance OIE methods to efficiently and effectively represent unstructured textual data?

The exponential growth of unstructured textual data has led to the emergence of Natural Language Processing (NLP)-based techniques that extract valuable information and insights. Information Extraction (IE) is a core NLP task that aids several applications. Nevertheless, existing IE systems lack two critical components: 1) a methodology capable of extracting valuable information, and 2) an appropriate data structure that facilitates efficient storage of the extracted knowledge. Therefore, in this thesis, we design an OIE framework that serves as the fundamental component for populating knowledge graphs. This leads to an efficient representation of textual data, which allows users to effectively query the information.

The research presented in this thesis showcases the effectiveness of utilizing contextual embeddings in the OIE process. Additionally, it demonstrates the benefits of employing data augmentation, multi-stage fine-tuning, and pre-trained language models in enhancing the efficiency and effectiveness of the IE process. Specifically, this dissertation proposes a two-stage fine-tuning procedure designed to strengthen the robustness of pre-trained language models for taxonomy classification tasks. Finally, we explore how OIE can be employed to construct knowledge graphs. We propose a framework, Open-CyKG, that achieves beyond-state-of-the-art results in the cybersecurity domain. Overall, this research demonstrates the potential of OIE systems and underscores the effectiveness of the proposed methodologies in extracting valuable knowledge from unstructured textual data

Universiteit Utrecht