Event Causality Extraction from Natural Language Text

Thesis submitted in partial fulfillment of the requirements for the degree of

Master of Engineering
in
Computer Science and Engineering

Submitted by

Arkadeep Sengupta

Regn. No. - 163566 of 2022-2023
Exam Roll No. - M4CSE24012
Class Roll No. - 002210502015
Session - 2022 - 2024

Under the supervision of

Dr. Sudip Kumar Naskar

Department of Computer Science and Engineering
Jadavpur University
188, Raja S.C. Mallick Rd,
Kolkata - 700032, West Bengal, India

Department of Computer Science and Engineering
Faculty of Engineering and Technology
Jadavpur University, Kolkata - 700032

DECLARATION OF ORIGINALITY AND COMPLIANCE OF ACADEMIC ETHICS

I hereby declare that the thesis entitled **"Event Causality Extraction from Natural Language Text"** contains literature survey and original research work by the undersigned candidate, as a part of his degree of **Master of Engineering** in **Computer Science and Engineering, Jadavpur University**. All the information have been obtained and presented in accordance with academic rules and ethical conduct.
I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work.

Name: Arkadeep Sengupta

Examination Roll No.: M4CSE24012

Registration No.: 163566 of 2022 - 2023

Thesis Title: Event Causality Extraction from Natural Language Text

Signature of the Candidate:

Department of Computer Science and Engineering
Faculty of Engineering and Technology
Jadavpur University, Kolkata - 700032

CERTIFICATE OF RECOMMENDATION

This is to certify that the work embodied in this thesis entitled **"Event Causality Extraction from Natural Language Text"** has been satisfactorily completed by **Arkadeep Sengupta** (Registration Number 163566 0f 2022-2023; Class Roll No. 002210502015; Examination Roll No. M4CSE24012). It is a bonafide piece of work carried out under my supervision and guidance at Jadavpur University, Kolkata, for partial fulfillment of the requirements for the awarding of the **Master of Engineering** in **Computer Science and Engineering** degree of the Department of Computer Science and Engineering, Faculty of Engineering and Technology, Jadavpur University, during the academic year 2022 - 2024.

Dr. Sudip Kumar Naskar,
Department of Computer Science and Engineering,
Jadavpur University.
(Supervisor)

Forwarded By:

Prof. Nandini Mukhopadhyay,
HOD, Department of Computer Science and Engineering,
Jadavpur University.

Prof. Dipak Laha,
DEAN, Faculty of Engineering and Technology,
Jadavpur University.

Department of Computer Science and Engineering
Faculty of Engineering and Technology
Jadavpur University, Kolkata - 700032

CERTIFICATE OF APPROVAL

This is to certify that the thesis entitled **"Event Causality Extraction from Natural Language Text"** is a bonafide record of work carried out by **Arkadeep Sengupta** (Registration No. 163566 of 2022 - 2023; Class Roll No. 002210502015; Examination Roll No. M4CSE24012) in partial fulfillment of the requirements for the award of the degree of **Master of Engineering** in **Computer Science and Engineering** in the Department of Computer Science and Engineering, during the period of September 2022 to May 2024. It is understood that by this approval, the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein, but approve the thesis only for the purpose of which it has been submitted.

Signature of Examiner 1
Date:

Signature of Examiner 2
Date:

Acknowledgements

I would like to express my sincere gratitude to my guide, **Dr. Sudip Kumar Naskar**, for his continuous motivation, guidance and support throughout the research journey. His expertise and dedication have been instrumental in completing my thesis.
I am also thankful to all other NLP lab members for their continuous support.

Arkadeep Sengupta

Examination Roll No.: M4CSE24012

Registration No.: 163566 of 2022 - 2023

Department of Computer Science and Engineering,

Jadavpur University.

Abstract

Extraction of events has been an important task in the field of Natural Language Processing. However, in the recent years establishing and studying of relationships between events has garnered wide importance. One such relationship between events is the cause-and-effect relationship which is present in major Socio-Political, Biomedical, Legal and Financial events and holds great importance. Hence detailed research on the causality between events has become so significant. Utilizing the Causal News Corpus (CNC) dataset, this thesis delves into the analysis of Socio-Political events worldwide, aiming to extract cause-effect spans to enhance our understanding of event causality. Leveraging the capabilities of Gen-AI (Llama-2), new data was generated and integrated into the CNC dataset, leading to a notable improvement in performance metrics. Specifically, the F1 Score for test data increased from 46% to 56% when employing the BERT model. Furthermore, to broaden the scope of the research, a comprehensive Causal News Corpus Bengali (CNCB) dataset was curated, encompassing news articles covering events from diverse geographical regions. This dataset, spanning over 100 news articles and consisting of 245 cause-effect relationships across 5 primary categories, serves as a valuable resource for studying event causality in Bengali. Additionally, by extracting the causes and effects from the CNCB dataset, a question answering dataset was also created called the CNCB-QA dataset, containing 355 questions in total—187 questions for the cause extraction subtask and 168 questions for the effect extraction subtask, respectively. Remarkable results were achieved through token classification and question answering tasks utilizing BERT as the underlying model, with an F1 score of 82.3% for token classification and 100% similarity score for the question answering task, highlighting the efficacy of the approach in extracting cause-effect spans. Overall, this research contributes to the growing field of event causality extraction, offering insights into the intricate relationships between events. The findings underscore the importance of causality studies in advancing our understanding of global phenomena and inform decision-making processes in various domains.

Contents

Abstract i

Table of Contents ii

List of Figures iv

List of Tables v

1 Introduction 1
- 1.1 Event Extraction . . . 1
- 1.2 Causality Extraction . . . 3

2 Literature Survey 6
- 2.1 Traditional Knowledge Based Methods . . . 6
- 2.2 Statistical Machine Learning Based Methods . . . 8
- 2.3 Deep Learning Based Methods . . . 10

3 Datasets 15
- 3.1 Causal News Corpus (CNC) Dataset . . . 15
- 3.2 Causal News Corpus Bengali (CNCB) Dataset . . . 17
- 3.3 CNCB Question Answering (CNCB-QA) Dataset . . . 18

4 Experiments and Results 20
- 4.1 Model Architecture . . . 20
  - 4.1.1 WordPiece Tokenizer . . . 20
  - 4.1.2 BERT Architecture . . . 22
    - 4.1.2.1 Embeddings . . . 22
    - 4.1.2.2 Encoder . . . 22
    - 4.1.2.3 Residual Connections and Layer Normalization . . . 23
- 4.2 Evaluation . . . 24
- 4.3 Causality Extraction from CNC Dataset . . . 25
  - 4.3.1 Experimental Setup . . . 26
  - 4.3.2 Results . . . 27
- 4.4 Causality Extraction from CNCB Dataset . . . 27
  - 4.4.1 Experimental Setup . . . 28
  - 4.4.2 Results . . . 28
- 4.5 Causality Extraction from CNCB-QA Dataset . . . 29
  - 4.5.1 Experimental Setup . . . 31

4.5.2 Results 31

5 Conclusions **32**
5.1 Conclusions of the Present Work 32
5.2 Scope for Future Works 33

Bibliography **34**

List of Figures

1.1 An example of two events in one sentence: Die and Attack. The upper arcs link event triggers to their corresponding arguments, with argument roles on the arcs. The lower side demonstrates the syntactic parser results. 2
1.2 An instance of event causality extraction. The cause event is marked in yellow and the effect event is marked in green. The cause-effect interactions include intra- and inter-event interactions. Solid lines indicate intra-event interactions between the event type and the corresponding arguments, while the dashed line indicates inter-event interaction between the cause-effect event. 4

2.1 Taxonomy of Causal Extraction techniques 7
2.2 The process to choose the appropriate technique for causality extraction 12

3.1 Augmentation Process for CNC Dataset 16
3.2 Examples illustrating the 5 major causal categories in CNCB Dataset; Causes in pink, Effects in green. 18
3.3 Examples illustrating the Context, Questions and Answers for the Cause and Effect Extraction Sub-tasks in CNCB-QA Dataset. . . . 19

4.1 Working of BERT Tokenizer 21
4.2 BERT Embedding Layer 22
4.3 BERT Architecture 23
4.4 Residual Connections 24
4.5 BERT for Token Classification Model Architecture 26
4.6 Examples showing input text labeled with ARG0 and ARG1 tags converted to BIO labels for Cause-and-Effect spans. 26
4.7 mBERT for Token Classification Model Architecture 28
4.8 Examples showing input text labeled with ARG0 and ARG1 tags converted to BIO labels for Cause-and-Effect spans. 28
4.9 mBERT for Question Answering Model Architecture 30
4.10 Flowchart for Predicted Answer Classification 30

List of Tables

3.1 CNC Dataset Statistics 16
3.2 CNCB Dataset Statistics 18
3.3 CNCB-QA Dataset Statistics 19

4.1 Results on the CNC Dataset 27
4.2 Results on the CNCB Dataset 29
4.3 Results on the CNCB-QA dataset 31

CHAPTER 1

Introduction

Event extraction is a natural language processing task that involves automatically identifying and extracting specific events and their relevant details from unstructured text. The process aims to recognize and extract essential information about events, such as the type of event, its participants, time, and location, transforming textual data into a structured format that can be analyzed more effectively.

On the other hand, Event causality delves into understanding the causal relationships between different events or entities mentioned in the text. It aims to identify cause-and-effect associations, revealing the underlying connections and dependencies among events. Event causality analysis helps unravel the reasons behind specific occurrences and sheds light on how different events are interconnected, providing valuable insights into the sequence and impact of events.

1.1 Event Extraction

An event is a specific occurrence of something that happens in a certain time and a certain place involving one or more participants, which can frequently be described as a change of state [1].

The aim of event extraction is to identify event instances within texts and determine the event components which are listed below:

- **Event mention:** An Event mention usually is a phrase or sentence that describes an event in which a trigger and corresponding arguments are included.

- **Event trigger**: It usually is a verb or a noun that most clearly expresses the core meaning of an event.

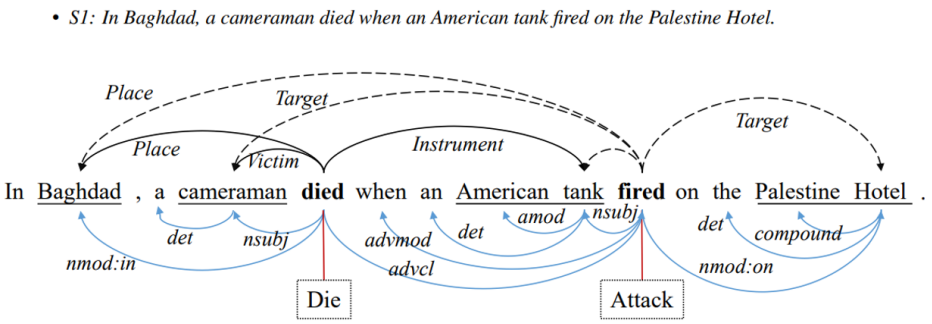


Figure 1.1: An example of two events in one sentence: Die and Attack. The upper arcs link event triggers to their corresponding arguments, with argument roles on the arcs. The lower side demonstrates the syntactic parser results.

- **Event argument**: Event arguments are the main attributes of events. They are usually entity mentions describing the event state change, involving who, what, when, where, and how.
- **Argument role:** An argument role is a function or position that an event argument performs in the relationship between the event argument and the trigger.

While event types may have varying arguments, the fundamental purpose of event extraction is to transform unstructured natural language into structured representations of events. This process helps answer the "5W1H" questions, which include "who, when, where, what, why," and "how," providing insights into real-world events from various textual sources such as news articles and social media posts. An example of event extraction is shown in Figure 1.1.

Event extraction plays an important role in various natural language processing applications. For example, Structured events obtained through event extraction serve as valuable additions to knowledge bases, enabling logical reasoning and inference. For governments, event detection and monitoring play a crucial role in public affairs management. By promptly identifying and understanding the developments of significant social events, authorities can respond efficiently and effectively.

In the business and financial sector, it aids companies in swiftly gauging market responses to their products and deriving signals for risk analysis and trading recommendations. In the biomedical domain, event extraction is employed to identify changes in bio-molecule states (e.g., genes and proteins) or interactions between bio-molecules, as described in scientific literature. This understanding of physiological and pathogenesis mechanisms proves invaluable. Overall, event extraction techniques and systems offer substantial benefits across multiple domains [2].

1.2 Causality Extraction

Events are often linked to temporal expressions, and their temporal attributes can be used to establish temporal relationships between events. This information is valuable for organizing event sequences in timelines, improving the presentation of news or historical texts [3].

A significant relationship between events is causality. Identifying causal relations between events is vital for predicting the occurrence of future events and can have substantial benefits in risk analysis and decision-making support. Listed below are the **Five Tests for Causality** given by Tan et al., 2022b [4] that helps in determining and extracting causal events from texts.

1. **Why:** The example is not causal if the reader is unable to construct a "Why" question regarding the Effect.
2. **Temporal order:** The example is not causal if the Cause does not precede the Effect in time.
3. **Counterfactual:** The example is not causal if the Effect is equally likely to occur or not occur without the Cause.
4. **Ontological asymmetry:** The example is not causal if the reader can readily swap the Cause and Effect claims in place.
5. **Linguistic:** The example is likely to be causal if it can be rephrased into "X causes Y" or "Due to X, Y."

Causality has been studied extensively in a wide range of disciplines, including Psychology, Linguistics, Philosophy and Computer Science. Causality mining (CM) in informal, ambiguous, heterogeneous, and domain-independent datasets has been a key topic in natural language processing (NLP) for decades, and it plays an important role in many related NLP applications including Question Answering , Event-Prediction, Scenarios Generation, Medical Text Decision, Machine Reading and Comprehension, Process Extraction [5]. The causality of events or the cause-effect relationship that exists between events holds three important properties which are as follows:

- **Temporal Sequence**: Cause and Effect events occur in a temporal sequence, where the cause precedes the effect in time.
- **Transitivity**: If A is the cause of B and B is the cause of C, then A indirectly causes C.

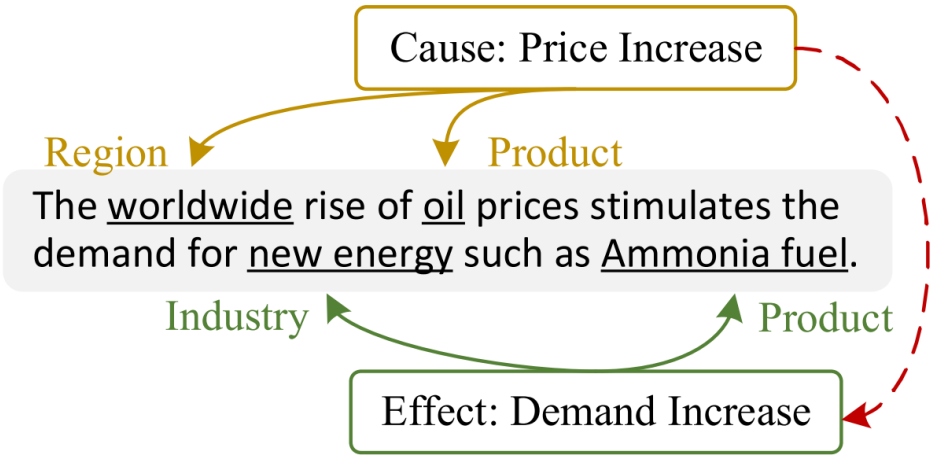


Figure 1.2: An instance of event causality extraction. The cause event is marked in yellow and the effect event is marked in green. The cause-effect interactions include intra- and inter-event interactions. Solid lines indicate intra-event interactions between the event type and the corresponding arguments, while the dashed line indicates inter-event interaction between the cause-effect event.

- **Asymmetry**: Cause and Effect relationships are asymmetric, meaning that the cause influences the effect but not vice versa.

The relation between event extraction and event causality is symbiotic. Event extraction provides the foundation for event causality analysis by identifying the events and their attributes that serve as the basis for establishing causal relationships. Event causality analysis, in turn, enriches event extraction by adding a layer of meaning to the extracted events. By combining these two processes, researchers can gain a deeper understanding of the complex narratives hidden within textual data. The integrated approach enhances our ability to construct comprehensive knowledge bases, enrich information repositories, and derive valuable insights from vast amounts of unstructured text data. Ultimately, the synergy between event extraction and event causality contributes significantly to advancing natural language understanding and knowledge discovery.

Causality extraction finds extensive applications across various domains. In finance and economics, causality extraction helps analysts and investors understand the relationships between different financial events and market movements. By extracting causal relationships from news articles, reports, and financial data, analysts can make more informed investment decisions and predict market trends with greater accuracy. This application is particularly valuable in high-frequency trading and algorithmic trading systems, where real-time insights into causal relationships can lead to competitive advantages.

In the healthcare domain, causality extraction plays a crucial role in pharmacy

co-vigilance and drug safety monitoring. By analyzing medical literature, patient records, and adverse event reports, researchers can extract causal links between drug interactions, side effects, and patient outcomes. This information is vital for identifying potential safety concerns, updating drug labels, and improving patient care by avoiding harmful drug combinations or dosages. Causality extraction also aids in post-marketing surveillance of pharmaceutical products, contributing to overall public health and drug regulatory efforts.

CHAPTER 2

Literature Survey

The task of causality extraction (CE) involves developing systems to identify cause-effect relationships between events in text. There are three main groups of approaches used in CE: knowledge-based, statistical machine learning based, and deep learning based methods. The study of CE can also be classified based on different representation patterns: explicit or implicit causality, intra-sentential or inter-sentential causality. Explicit causality involves clear cause-effect relationships connected by explicit causal connectives like words such as "so," "hence," "therefore," and phrases like "because of" or "on account of." Causative verbs (e.g., break, kill), resultative phrases, conditionals (i.e., if...then...), and causative adverbs and adjectives also indicate explicit causality. Implicit causality, on the other hand, lacks explicit causal cues, requiring readers to use background knowledge for analysis and reasoning to identify causality in the text. In intra-sentential causality, both the cause and effect are within a single sentence, while in inter-sentential causality, the cause and effect are present in different sentences. Many texts contain implicit and/or inter-sentential causality, which is more complex than the basic forms of causality.

This literature survey explores various approaches, techniques, and challenges in this domain, aiming to provide insights into the current state-of-the-art methods and future research directions.

2.1 Traditional Knowledge Based Methods

Knowledge-based causal extraction (CE) systems can be categorized into pattern based and rule-based approaches. Pattern-based systems use pre-defined graphical patterns or keywords to express linguistic patterns, and they can also explore patterns through sentence structure analyses for better performance and of implicit causality relations. On the other hand, rule-based systems may rely on

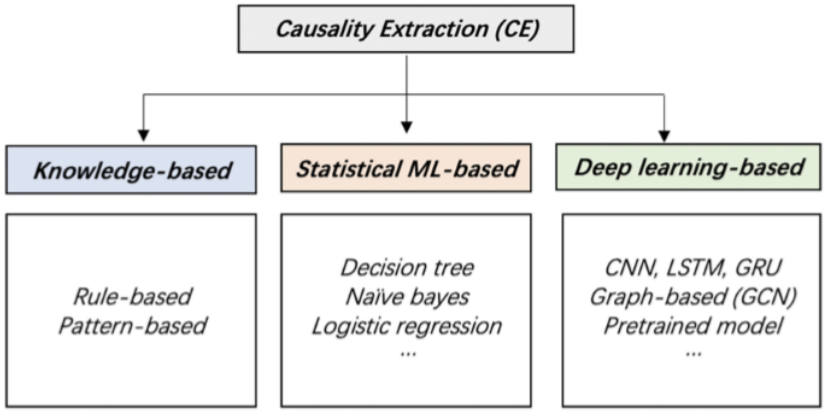


Figure 2.1: Taxonomy of Causal Extraction techniques

patterns or templates like pattern-based approaches, but others utilize heuristic algorithms on sentence syntactic structures.

Patterns are used to identify explicitly expressed causal relations within single sentences. COATIS, developed by Garcia et al. [6] extracts causality from French texts through lexico-syntactic patterns based on explicit causal verbs, achieving 85.2% precision.

Radinsky et al. [7] propose a rule-based Pundit algorithm for generating causality pairs from news articles, achieving higher automation but with limited recall.

Ittoo and Bouma [8] present a minimally supervised method for identifying three types of implicit causality iteratively. They define causal patterns, including resultative verbal patterns with verbs like increase, reduce, kill, become; inseparable cause-effect patterns; and nonverbal patterns like rise and due to, which are acquired from Wikipedia. Their experimental study in the Product Development Customer Service domain achieves an 85.0% F-score, comparable to state-of-the-art systems.

In another study, Kang et al. [9] focus on extracting adverse drug effects from the ADE corpus using a concept identification module and a knowledge-based module that checks for relationships between concepts. The system combines a rule-based NLP module with a dictionary-based concept recognition and normalization tool called Peregrine for recognizing relevant concepts.

In their approach, Khoo et al. [10] utilize four types of causative links and 2082 causative verbs to construct verbal linguistic causal extraction (CE) patterns. Their computer program can identify causal relations within a sentence and between adjacent sentences by matching these linguistic patterns. However, the system achieves an accuracy of 68.0% on a set of 1082 sentences from the WSJ newspaper, with errors attributed to complex sentence structures and the lack of

inferencing capability from world knowledge.

2.2 Statistical Machine Learning Based Methods

Statistical machine learning-based approaches in causal extraction require fewer manual pre-defined patterns compared to knowledge-based approaches. These methods make use of third-party natural language processing (NLP) tools like Spacy, Stanford CoreNLP or Stanza to generate a set of features from labelled data. After extracting the relevant features, machine learning algorithms such as support vector machine (SVM), maximum entropy (ME), naïve Bayes (NB), and logistic regression (LG) are employed for classification tasks. These statistical machine learning techniques offer a detailed and effective approach for causality detection and identification rather than causality extraction.

Various models for detecting causal relations have been proposed. Girju's [11] model introduces a method for the automatic detection and extraction of causal relations through pattern matching. It also presents an inductive learning approach to discover lexical and semantic constraints that help disambiguate these causal relations. A classification system for causal questions was developed and tested on a QA system to evaluate the effectiveness of this approach.

Blanco et al. [12] presents a supervised method for detecting and extracting causal relations from open domain text. It begins with an overview of causation, its relation to other semantic relations, and how causation is encoded. The focus is on marked and explicit causations. The approach involves identifying syntactic patterns that may indicate causation and then using machine learning techniques to confirm if a pattern instance encodes causation. The primary patterns examined are a verb phrase followed by a relator and a clause, and its reverse. Relators include the words "as", "after", "because" and "since." The paper details the lexical, syntactic, and semantic features used for classification, provides their rationale and examples, and discusses the results and error analysis. This approach achieves a high F-score of 91.3% on a question classification dataset.

Pakray and Gelbukh [13], introduces a supervised method for detecting causal relations from text, leveraging dependency relations, WordNet features, and Parts-of-Speech (POS) features. Using the SemEval-2010 Task 8 datasets, with 7954 training instances and 2707 testing instances, the method employs the decision tree algorithm to identify semantic causal relations between nominal pairs. The approach achieved an overall F1 score of 85.8% for detecting causal instances.

Sorgente et al. [14] presents a hybrid approach for discovering causal relations from text, combining rule-based methods and machine learning techniques. Initially, logical rules identify plausible cause-effect pairs based on word dependencies.

Bayesian inference is then used to filter out ambiguous pairs. The approach was evaluated using the SemEval-2010 Task 8 dataset, demonstrating the effectiveness of the rule-based extraction and the improvements from the filtering process.

Zhao et al. [15] introduces a new Restricted Hidden Naive Bayes model for extracting causality from texts. In addition to common features such as contextual, syntactic, and positional features, the model uses a novel category feature for causal connectives derived from the tree kernel similarity of sentences containing these connectives. Unlike previous studies that often assume feature independence, this model addresses partial interactions among features to prevent overfitting, particularly the interaction between connective categories and syntactic structures. Evaluation on the SemEval-2010 Task 8 corpus demonstrates that this method surpasses all existing baselines.

Hidey and McKeown [16] address the shortage of causal connectives by using PDTB 2.0 AltLex connectives as seed data to identify new alternative lexicalizations from parallel corpora. They train an SVM classifier with parallel connectives and lexico-semantic features, achieving an F-score of 75.3%, an 11.1% improvement over the baseline on the AltLex corpus. Airola et al. [17] use a dependency path kernel to extract protein-protein interactions from BioInfer, achieving success with kernel-based machine learning. Keskes et al. [18] propose a ME model forlearning causality in Arabic, achieving an F-score of 78.1% and accuracy of 80.6% with eight linguistic features. Pechsiri et al. [19] use verb-pair rules, NB, and SVM to mine implicit causality from Thai texts, obtaining precision and recall metrics of 86.0% and 70.0%, respectively. Gurulingappa and colleagues [20] train a ME model with simple features to signal the availability of the ADE corpus, serving as baseline performance values for other systems.

Marcu and Echihabi [21] use lexical pair probability to discern causality in inter-sentential forms. They identify candidate sentence pairs using keywords "Because of" and "Thus" and explore causal lexical pairs with pre-collected explicit causality nouns, verbs, and adverbs.

Oh et al. [22] develop a Japanese why-QA system that identifies cue phrases for causality using regular expressions with explicit keywords, then extracts three sentences as one causality candidate for each cue phrase. They use semantic and syntactic features with a conditional random field (CRF) model to generate cause-effect labels for each word.

Qian and Zhou [23] propose two ME models to extract chemical-induced disease (CID) relations from biomedical articles at both intra- and inter-sentential levels. They use heuristic rules to construct training and test instances at the inter-sentence level and employ an ME classifier with lexical features to extract this relationship from a collection of 1500 medical articles.

2.3 Deep Learning Based Methods

Neural networks (NNs) are fundamental algorithms in deep learning (DL) that mimic the human neural system. They consist of three types of layers: input, hidden, and output. Neurons in each layer receive input from previous neurons and produce output for subsequent neurons. When an NN learns multiple levels of representation through multiple hidden layers, it becomes a "deep" neural network, and this process is called "deep learning."

Compared to knowledge-based and statistical machine learning models, deep learning models transform words and features into low-dimensional dense vectors, which helps mitigate the feature sparsity problem. Additionally, the use of attention mechanisms allows these models to focus on relevant aspects while ignoring others, making deep learning models more effective. Since deep learning can automatically deduce higher-level information from input vectors and make adjustments to the expected results, these systems are able to focus more on the choice of input features and model architecture, rather than on the preparation of linguistic information. Deep learning-based systems thus have better portability to different applications. However,they require access to larger corpora and more substantial computational resources than the other two techniques. As illustrated in Fig. 2.2, stated in the survey by Yang et al. [24] specific needs and other contextual aspects should be taken into consideration in order to choose an appropriate method for causality extraction.

Common deep learning models include convolutional neural networks (CNNs) and recurrent neural networks (RNNs). Variants of RNNs such as long short-term memory (LSTM) and gated recurrent units (GRU) are also widely used. Unsupervised pretraining language models (PTMs) like BERT have significantly improved performance on various natural language processing (NLP) tasks by providing contextualized embeddings for each token.

CNNs and RNNs are sequential-based models, capturing semantic and syntactic information in local consecutive word sequences. In contrast, graph-based models such as graph convolutional networks (GCNs) and graph attention networks (GATs) model relationships between nodes in a graph, allowing them to handle more complex structures. In the following paragraphs, the text discusses how deep learning architectures have been applied to solve the Causality Extraction problem in NLP.

The below mentioned studies are dedicated to utilizing different deep learning models to extract causality. Xu et al. [25] introduces SDP-LSTM, an innovative neural network for relation classification in NLP, which leverages the shortest dependency path (SDP) between entities and utilizes multi-channel recurrent

neural networks with LSTM units to gather diverse information along the SDP. Key features include the retention of relevant information through SDP, effective information integration from various sources via multi-channel LSTMs, and a customized dropout strategy to mitigate overfitting. Tested on the SemEval 2010 relation classification task, SDP-LSTM achieved an impressive F1-score of 83.7%, surpassing existing methods.

Li et al. [26] redefines causality extraction as a sequence labeling task using a novel tagging scheme and introduces SCITE (Self-attentive BiLSTM-CRF wIth Transferred Embeddings). SCITE, which utilizes the BiLSTM-CRF model, directly extracts causes and effects without the need to separately identify candidate causal pairs and their relationships. To mitigate data insufficiency, the model incorporates Flair embeddings trained on a large corpus. Additionally, a multihead self-attention mechanism is integrated to enhance the learning of dependencies between causal words. Evaluation on a public dataset shows that SCITE significantly outperforms existing baseline methods.

Kyriakakis et al. [27] explore the application of pretraining language models (PTMs) like BERT and ELMO for causality extraction using bidirectional GRU with self-attention (BIGRUATT) as the baseline. PTMs prove useful for datasets with hundreds of training examples, and BIGRUATT reaches a performance plateau with thousands of instances.

Martínez-Cámara et al. [28] propose an LSTM model that uses only word embeddings as input, avoiding the use of linguistic features to represent causality. The experimental results on the PDTB 2.0 AltLex corpus show a promising F-score of 81.9%.

Joint models for entity and relation extraction from the ADE corpus are proposed by Li et al. [29], Wang and Lu [30], Zhao et al. [31].

Li et al. [29] proposed a neural joint model that simultaneously extracts biomedical entities and their relations. Evaluated on two tasks—extracting adverse drug events between drug and disease entities, and extracting resident relations between bacteria and location entities—the model significantly outperformed state-of-the-art systems, improving F1 scores by 5.1% for entity recognition and 8.0% for relation extraction in the first task, and by 9.2% for relation extraction in the second task. The results show that neural networks are effective for biomedical entity and relation extraction, reducing the need for feature engineering and facilitating biomedical text mining research.

Wang and Lu [30] proposed a novel approach with table-sequence encoders, utilizing two distinct encoders—a table encoder and a sequence encoder—to separately and effectively capture the different types of information required for NER and relation extraction. Experimental results demonstrate that having two spe-

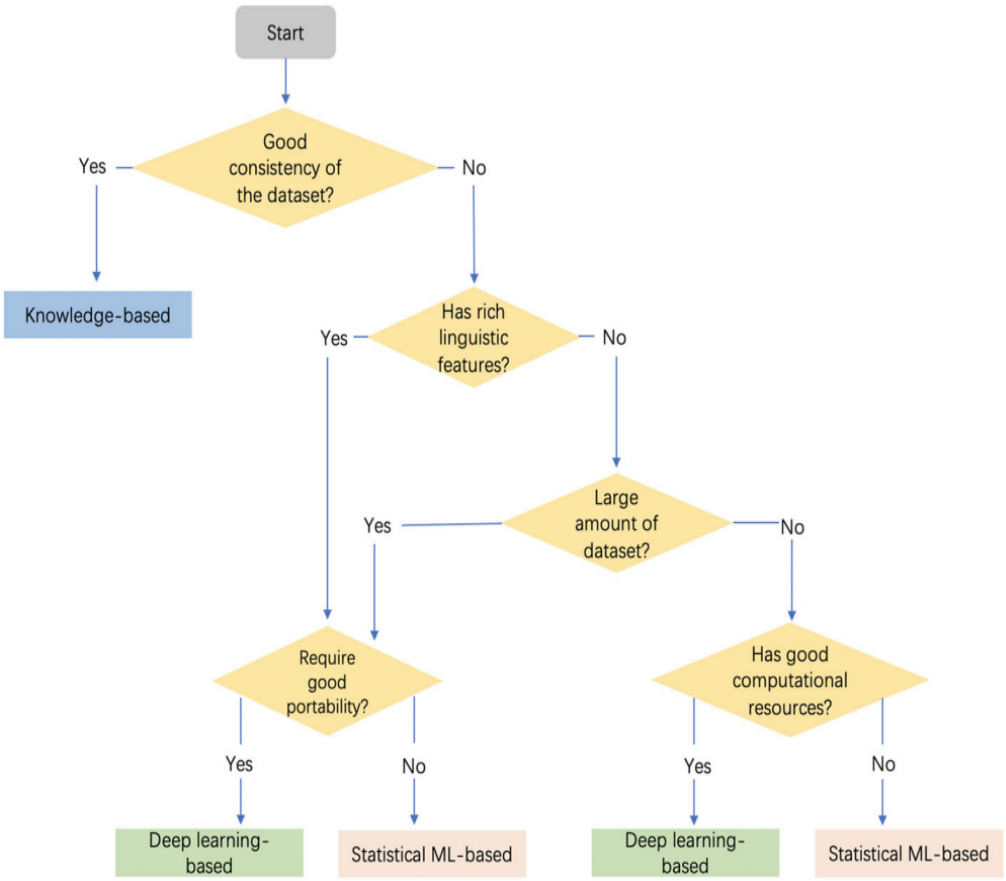


Figure 2.2: The process to choose the appropriate technique for causality extraction

cialized encoders provides significant improvements over using a single encoder, achieving better performance on several standard datasets.

Zhao et al. [31] construct a cross-modal attention network (CMAN) using BiLSTM-enhanced self-attention (BSA) and BiLSTM enhanced label-attention (BLA) units to obtain dense correlations over token and label spaces, achieving state-of-the-art performance on ADE with an F-score of 81.1%.

Dasgupta et al. [32] extend the annotation of the SemEval 2010 Task 8 to the phrase-level and propose a linguistically informed BiLSTM model that encodes word embeddings with linguistic features. The misclassification of causal instances as non-causal instances is attributed to the dependency parser's failure to parse the texts correctly, leading to improperly syntactic features.

In the recent times studying causality in socio-political events have gained great importance. The study of causality in socio-political events may involve both retrospective analysis and prospective forecasting. By identifying causal relationships and underlying patterns, researchers can gain insights into the drivers of past events and potentially anticipate future developments.

Kim et al. [33] employed a data augmentation technique utilizing Part-Of-

Speech (POS) based on the observation that some parts of speech are more (or less) relevant to causality. The approach which used the ELECTRA model as the base model especially improved the recall of detecting causal events in sentences.

The majority of current systems for socio-political event detection rely on text-based data, yet a growing volume of web content is multimodal. To address this gap, Raj et al. [34] propose a method that utilizes annotated unimodal data to perform event detection in another modality, specifically in a zero-shot manner. Our focus is on protest detection in both text and images, and we demonstrate that a pretrained vision-and-language alignment model, such as CLIP, can be effectively leveraged for this purpose. Our findings indicate that annotated protest text data can complement the detection of protests in images, while substantial transferability is also demonstrated in the reverse direction. This approach demonstrates promise for bridging the gap between text-based and multimodal event detection systems.

Fajcik et al. [35] detect cause-effect-signal spans in a sentence using T5 a pre-trained autoregressive language model. They iteratively identify all cause-effect-signal span triplets, always conditioning the prediction of the next triplet on the previously predicted ones. To predict the triplet itself, they consider different causal relationships such as cause→effect→signal. Each triplet component is generated via a language model conditioned on the sentence, the previous parts of the current triplet, and previously predicted triplets.

Trust et al. [36] aimed to detect causal relationships between event mentions in text. While deep learning models have achieved state-of-the-art performance in various NLP tasks, they often struggle to capture the rich semantic and syntactic structures within sentences essential for causality classification. The proposed solution leverages gated graph neural networks (GGNN) and contextualized language embeddings to capture this crucial information.

Chen et al. [37] model the task as a reading comprehension (RC) problem and apply a token-level RC-based span prediction paradigm to the task as the baseline. They explore different training objectives to fine-tune the model, as well as data augmentation (DA) tricks based on the language model (LM) for performance improvement. Additionally, they propose an efficient beam-search post-processing strategy to due with the drawbacks of span detection to obtain a further performance gain.

Akdemir et al. [38] tackle the classification problem by framing it as an entailment problem and employing zero-shot ranking on socio-political texts. This approach allows for the top-ranked documents to be considered positively classified, thus reducing the close reading time needed for information extraction. We utilize Transformer Language Models to obtain entailment probabilities and ex-

plore various query types. Our experiments reveal that DeBERTa achieves higher mean average precision scores than RoBERTa, and using the declarative form of the class label as a query outperforms the dictionary definition of the class label. We demonstrate that the close reading time can be reduced by selecting a certain percentage of top-ranked documents, contingent on the desired recall level. However, our results also indicate that the percentage of documents requiring close reading increases as the topic broadens.

Saha et al. [39] address the causal information extraction problem with different neural models built on top of pre-trained transformer-based language models for identifying Cause, Effect and Signal spans, from news data sets. They use the Causal News Corpus subtask 2 training data set to train span-based and sequence tagging models. The span-based model based on pre-trained BERT base weights achieves an F1 score of 47.48% on the test set with an accuracy score of 36.87%.

Tan et al. [40] created Resource for Extracting Cause, Effect and Signal Spans (RECESS), a comprehensive corpus annotated for causality at different levels, including Cause, Effect, and Signal spans. The work also focuses on how causal relations are related to causal question answering (QA). To adapt causal relations for QA, one can create questions from the cause or effect and use the corresponding argument as the answer. For example, asking "Why did <effect>?" with the cause as the answer. Using this method, 4,514 Why-Questions were generated from causal relations in the training set with two randomly selected templates.

The baseline model, t5-small, was trained on the train set and evaluated on the development set of SQuAD. An improved version of this model was pre-trained on the generated Why-Questions before following the same training schedule. In the evaluation, 335 Why-Questions containing the words 'reason', 'why', 'cause', or 'resulted' were identified. The baseline model scored 53.42 EM and 63.21 F1, while the improved model scored higher with 55.52 EM and 65.61 F1 for Why-Questions. This indicates that pre-training on RECESS, which focuses on causal questions, enhances performance on Why-Questions in QA tasks.

CHAPTER 3

Datasets

Datasets play a crucial role in any research. Firstly, they serve as the foundation for training and validating machine learning or deep learning models. Moreover, datasets enable researchers to explore hypotheses, test theories, and derive insights from real-world data. They provide empirical evidence to support findings and conclusions, enhancing the credibility and robustness of research outcomes.

For the purpose of event causality extraction, we used the Causal News Corpus (CNC) dataset annotated with causal information in English. Also, to broaden the scope of research, we curated a comprehensive CNC Bengali (CNCB) dataset similar to the CNC dataset. The CNCB-QA dataset, derived from the CNCB dataset, serves as a question answering dataset designed to extract cause-and-effect spans through question-answering.

3.1 Causal News Corpus (CNC) Dataset

The corpus we used in our model training and evaluation is the CNC dataset (Tan et al., 2022b) [4]. This dataset is built on the extraction of social-political events from News (AESPEN) (Hürriyetoglu et al., 2020) [41] in 2020 and the CASE 2021 workshop @ ACL-IJCNLP (Hürriyetoglu et al., 2021) [42]. Each sample in the dataset is annotated with causal labels, that is, whether a sentence contains a causal event. Furthermore, sentences are annotated with the span of the specific Cause and Effect of a causal event, that imply the causality. The spans are labelled by <ARG0> and <ARG1> annotations to represent the cause-and-effect spans in the sentence, respectively. Note that it is possible to have multiple annotations for the same sentence in the dataset if the sentence contains multiple casual relationships of events.

The CNC dataset consists of 3767 sentences out of which 1982 sentences have causal relations and the rest 1785 are non-causal. The 1982 sentences having causal

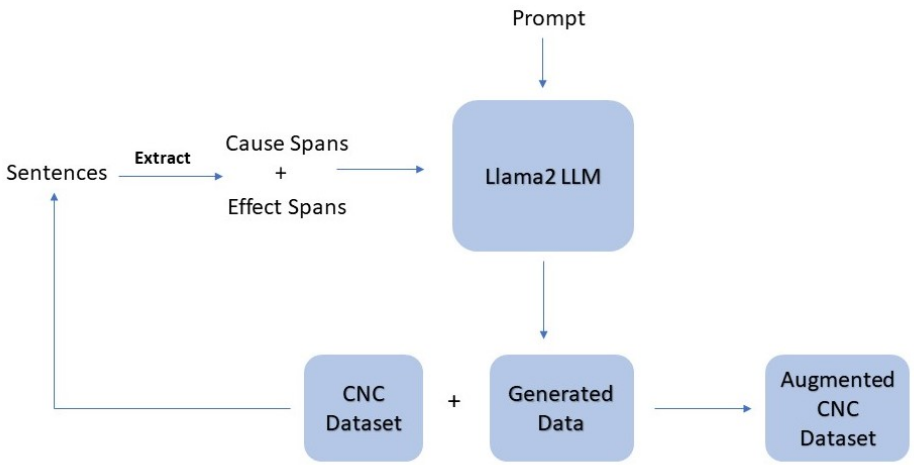


Figure 3.1: Augmentation Process for CNC Dataset

CNC Dataset	**Statistic**	**Train**	**Dev**	**Test**	**Total**
Original	# Sentences	1624	185	173	1982
	# Relations	2257	249	248	2754
Updated	# Sentences	1444	185	180	1809
	# Relations	2007	249	250	2506
Generated using LLaMa 2	# Sentences	56	0	0	56
	# Relations	103	0	0	103
Augmented	# Sentences	1500	185	180	1865
	# Relations	2110	249	250	2609

Table 3.1: CNC Dataset Statistics

relations are divided into train, dev and test data with 1624, 185 and 173 sentences respectively. However, the 173 sentences in the test data are not annotated. Hence the 1809 causal sentences from the train and dev are integrated and divided into three groups forming the train, dev and test datasets for our experiment.

The CNC dataset is further augmented with causal sentences generated from the LLaMa 2 (Llama-2-7b-chat-hf) [43] language model. Cause and Effect spans are extracted from the sentences of the CNC dataset which are the fed into the LLaMa 2 language model along with a suitable prompt to generate new sentences but with the same cause and effect spans. The generated sentences are then augmented to the CNC dataset for further experiments. The entire process of data augmentation is shown in Figure 3.1. 56 sentences with 103 causal relationships are augmented to the above training dataset. The original, updated and augmented CNC dataset statistics is shown in Table 3.1.

3.2 Causal News Corpus Bengali (CNCB) Dataset

The Causal News Corpus Bengali (CNCB) dataset serves as a Bengali corpus with the primary aim of investigating event causality extraction within Bengali text. Despite the pivotal role of event causality detection in NLP, there is a glaring scarcity of corpora available for Indian languages. Hence, a key objective of this thesis is to develop a Bengali dataset, which could serve as a starting point for exploring, identifying, and comprehending the basis of event causality across Indian languages.

The CNCB dataset spanning over 100 news articles and consisting of 245 cause-effect relationships across 5 primary categories, serves as a valuable resource for studying event causality in Bengali. Similar to the CNC dataset, as shown in the Figure 3.2, the CNCB dataset is annotated with the span indicating the specific cause and effect of a causal event, aligning with one of the five major categories of causality outlined below. The spans are labelled by <ARG0> and <ARG1> annotations to represent the cause-and-effect spans in the sentence, respectively. The detailed statistics of CNCB dataset is shown in Table 3.2.

The 5 major categories of cause-effect relationships between events are listed below:

- **Explicit** - Single Cause and Single Effect which are present in the same sentence having clear and direct relationship between two events where one event directly influences or causes the occurrence of another event.
- **Implicit** - Single Cause and Single Effect which are not present in the same sentence having indirect relationship between events where the cause-and-effect link may not be immediately apparent or explicitly stated.
- **Chained** - A chain or sequence of events where each event (except the first and the last event) serves as both the effect of the preceding event and the cause of the subsequent one, forming a continuous chain of causation. The first event is only a cause, the first cause that starts the entire chain of events and the last event is only an effect, the final effect that concludes the chain of causation.
- **One Cause Multiple Effect (OCME)** - An event which serves as a major cause that triggers a list of potential outcomes or effects.
- **Multiple Cause One Effect (MCOE)** - A list of events each serving as a potential cause leading to a singular effect.

	All	Train	Validation	Test
No of Articles	100	80	10	10
Avg Sentences/ Article	3.68	3.93	2.5	2.8
Total Relations	245	214	17	14
Combined Relations	186	162	12	12
Total Exp Relations	93	84	5	4
Total Imp Relations	51	42	3	6
Total Chained Relations	12	9	2	1
Total OCME Relations	24	22	2	0
Total MCOE Relations	6	5	0	1

Table 3.2: CNCB Dataset Statistics

Sentences	Type	Span Annotations
মেয়ের আত্মহত্যার খবর শুনে হাতের শিরা কেটে নিজেকে শেষ করে দেওয়ার চেষ্টা করলেন বৃদ্ধা মা।	Explicit	<ARG0> মেয়ের আত্মহত্যার খবর শুনে </ARG0> <ARG1> হাতের শিরা কেটে নিজেকে শেষ করে দেওয়ার চেষ্টা করলেন বৃদ্ধা মা </ARG1>।
অস্থায়ী কর্মীদের বেতন নেই দীর্ঘদিন। অসহায়তায় দিন কাটাচ্ছেন তাঁরা।	Implicit	<ARG0> অস্থায়ী কর্মীদের বেতন নেই দীর্ঘদিন </ARG0>। <ARG1> অসহায়তায় দিন কাটাচ্ছেন তাঁরা </ARG1>।
গরু চুরির চেষ্টার অভিযোগে গণপ্রহার। আর তাতে মৃত্যু হল দু'জনের।	Chained	<ARG0> গরু চুরির চেষ্টার অভিযোগে </ARG0> <ARG1> গণপ্রহার </ARG1>। আর তাতে মৃত্যু হল দু'জনের।
		গরু চুরির চেষ্টার অভিযোগে <ARG0> গণপ্রহার </ARG0>। আর <ARG1> তাতে মৃত্যু হল দু'জনের </ARG1>।
পুরনো ল্যান্ডমাইনের উপর পা দিতেই বিপত্তি। প্রাণ হারালেন এক জওয়ান। আহত আরও দু'জন।	OCME	<ARG0> পুরনো ল্যান্ডমাইনের উপর পা </ARG0> দিতেই বিপত্তি। <ARG1> প্রাণ হারালেন এক জওয়ান </ARG1>। আহত আরও দু'জন।
		<ARG0> পুরনো ল্যান্ডমাইনের উপর পা </ARG0> দিতেই বিপত্তি। প্রাণ হারালেন এক জওয়ান। <ARG1> আহত আরও দু'জন </ARG1>।
এক দিকে ভিন রাজ্য থেকে জোগান কমে যাওয়া। অন্য দিকে অতিরিক্ত চাহিদার চাপ- দুইয়ে মিলে হঠাৎই মহার্ঘ্য হয়ে উঠেছে ডিম।	MCOE	এক দিকে <ARG0> ভিন রাজ্য থেকে জোগান কমে যাওয়া </ARG0>। অন্য দিকে অতিরিক্ত চাহিদার চাপ- দুইয়ে মিলে হঠাৎই <ARG1> মহার্ঘ্য হয়ে উঠেছে ডিম </ARG1>।
		এক দিকে ভিন রাজ্য থেকে জোগান কমে যাওয়া। অন্য দিকে <ARG0> অতিরিক্ত চাহিদার চাপ </ARG0> - দুইয়ে মিলে হঠাৎই <ARG1> মহার্ঘ্য হয়ে উঠেছে ডিম </ARG1>।

Figure 3.2: Examples illustrating the 5 major causal categories in CNCB Dataset; Causes in pink, Effects in green.

3.3 CNCB Question Answering (CNCB-QA) Dataset

The Causal News Corpus Bengali Question Answering (CNCB-QA) dataset is an extension of the CNCB dataset created specifically for the extraction of cause-and-effect spans through question answering. The creation of the dataset was inspired from the works of Tan et al. [40] where the causality extraction task was reduced to a QA task inorder to address the question, "How are causal relations related to causal question answering (QA)?"

The dataset is divided into two parts QA for cause extraction and QA for effect extraction. However, the combined CNCB-QA dataset can also be used for Question Answering (QA) task in Bengali.

The CNCB-QA dataset is created by extracting the causes and effects from

Sub-Task	**Total QA**	**Train**	**Validation**	**Test**
Effect Extraction	**187**	155	17	15
Cause Extraction	**168**	140	16	12
Total	**355**	295	33	27

Table 3.3: CNCB-QA Dataset Statistics

Sub-Task	Context	Cause	Effect	Question	Answers
Cause Extraction	মেয়ের আত্মহত্যার খবর শুনে হাতের শিরা কেটে নিজেকে শেষ করে দেওয়ার চেষ্টা করলেন বৃদ্ধা মা।	মেয়ের আত্মহত্যার খবর শুনে	হাতের শিরা কেটে নিজেকে শেষ করে দেওয়ার চেষ্টা করলেন বৃদ্ধা মা	কেন হাতের শিরা কেটে নিজেকে শেষ করে দেওয়ার চেষ্টা করলেন বৃদ্ধা মা?	মেয়ের আত্মহত্যার খবর শুনে
Effect Extraction	দুই অঙ্গনওয়াড়ি কেন্দ্রে চাল নেই, চার দিন রান্না বন্ধ।	দুই অঙ্গনওয়াড়ি কেন্দ্রে চাল নেই	চার দিন রান্না বন্ধ	দুই অঙ্গনওয়াড়ি কেন্দ্রে চাল না থাকার জন্য কি হলো?	চার দিন রান্না বন্ধ

Figure 3.3: Examples illustrating the Context, Questions and Answers for the Cause and Effect Extraction Sub-tasks in CNCB-QA Dataset.

the CNCB dataset and creating questions from the effect to extract the cause and vice versa where the original article text serves as the context. An example of the same is shown in Figure 3.3. There are 355 questions in total out of which 187 questions are for the effect extraction subtask and 168 questions are for the cause extraction subtask. The statistics for the CNCB-QA dataset is shown in Table 3.3.

CHAPTER 4

Experiments and Results

In this study, we conducted experiments on three distinct datasets: the CNC dataset, CNCB dataset, and CNCB-QA dataset, with the aim of extracting various cause-and-effect events. Our experimental approach is centered on leveraging the encoder-only transformer BERT model to grasp contextual information, subsequently generating contextual embeddings for sequence tagging and question-answering tasks, which are integral to our experiments.

4.1 Model Architecture

The model architecture consists mainly of the WordPiece tokenizer, and the BERT model designed to understand the contextual relationships between words in a sentence which in turn helps in the extraction of cause-effect spans.

4.1.1 WordPiece Tokenizer

The WordPiece tokenizer which serves as the foundation for the BERT Tokenizer [44] is used to tokenize raw text and preprocess it to convert it into a format that can be fed into the BERT (Bidirectional Encoder Representations from Transformers) model. An illustration depicting the working of the BERT Tokenizer is shown in Figure 4.1. The major functions done by the BERT Tokenizer are listed below:

1. **Tokenize using the Wordpiece Tokenizer:** The Wordpiece Tokenizer is a subword tokenizer employed by BERT to segment words into smaller units called subwords or tokens. This tokenizer divides input text into meaningful subword units, enabling BERT to handle out-of-vocabulary words more effectively. Each subword token is represented by its corresponding index in the BERT vocabulary.

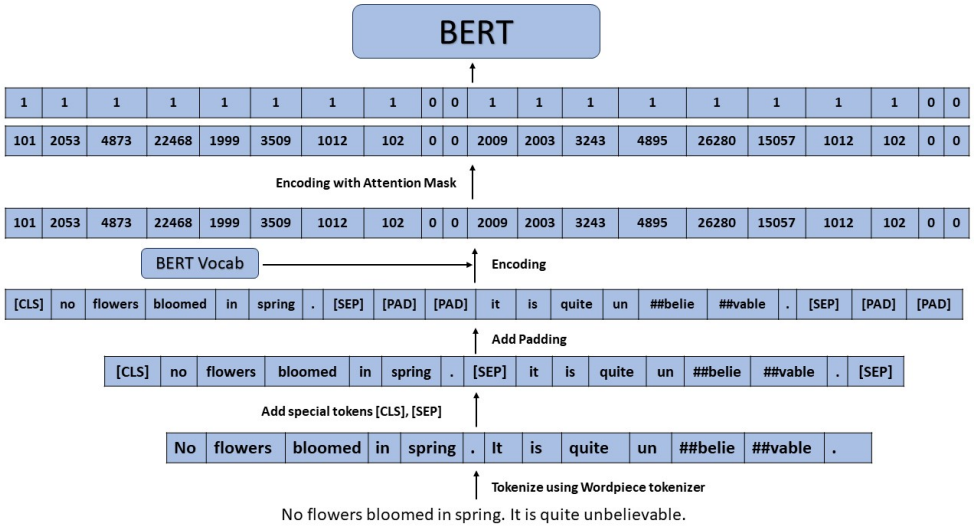


Figure 4.1: Working of BERT Tokenizer

2. **Adding special tokens**: BERT requires special tokens to denote the beginning ([CLS]) and separation ([SEP]) of sentences or text segments. [CLS] is placed at the beginning of the input text, indicating the classification task, while [SEP] is used to separate two sentences or text segments.

3. **Add Padding:** BERT requires inputs to be of fixed length. If the input text is shorter than the maximum sequence length allowed by BERT, the BERTTokenizer adds padding tokens (usually represented by the index 0) to make all input sequences of equal length. Padding ensures uniformity in input dimensions across different samples, facilitating batch processing.

4. **Encoding:** After tokenization and padding, the BERTTokenizer converts the tokens into numerical representations. Each token is mapped to its corresponding index in the BERT model's vocabulary. This process turns the input text into a sequence of integers that can be fed into the BERT model for further processing.

5. **Add Attention Mask:** The attention mask decides which tokens are to be considered and which are not to be considered by the attention layer in the BERT model: The attention mechanism in BERT attends to different parts of the input sequence to capture contextual information effectively. The attention mask is a binary mask indicating which tokens are actual input tokens (1) and which are padding tokens (0). By masking padding tokens, the attention layer focuses only on relevant tokens during processing, improving computational efficiency and model performance.

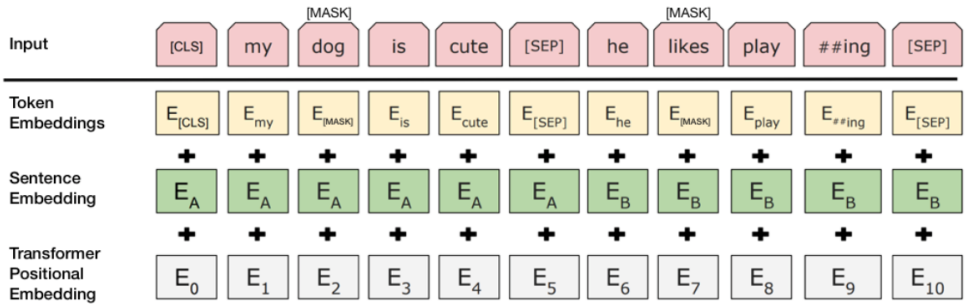


Figure 4.2: BERT Embedding Layer

4.1.2 BERT Architecture

BERT (Bidirectional Encoder Representations from Transformers) [45] is a state-of-the-art natural language processing model developed by Google. It utilizes a transformer architecture, which is a type of neural network architecture known for its effectiveness in capturing long-range dependencies in sequential data like text. Listed below are the components that make up the BERT architecture.

4.1.2.1 Embeddings

The first layer in the BERT model is the embedding layer, shown in Figure 4.2, which mainly consists of three types of embeddings that captures the semantic and positional information of the input text.

1. **Token Embedding:** BERT takes input text and converts each token (word or subword) into a high-dimensional vector representation. This token embedding captures the semantic meaning of each word in the context of the entire sentence.

2. **Sentence Embedding:** Sentence embedding in BERT is a crucial component designed to help the model understand and differentiate between multiple sentences or segments within a single input sequence.

3. **Positional Embedding:** Since transformers do not inherently understand the order of tokens, BERT incorporates positional embedding to encode the position of each token in the sequence. This positional information is crucial for understanding the sequential nature of language.

4.1.2.2 Encoder

The Encoder is the main component of the BERT architecture responsible for capturing the contextual information of the text with the help of the Attention mechanism. Based on the BERT size layers of encoders are stacked on top of each

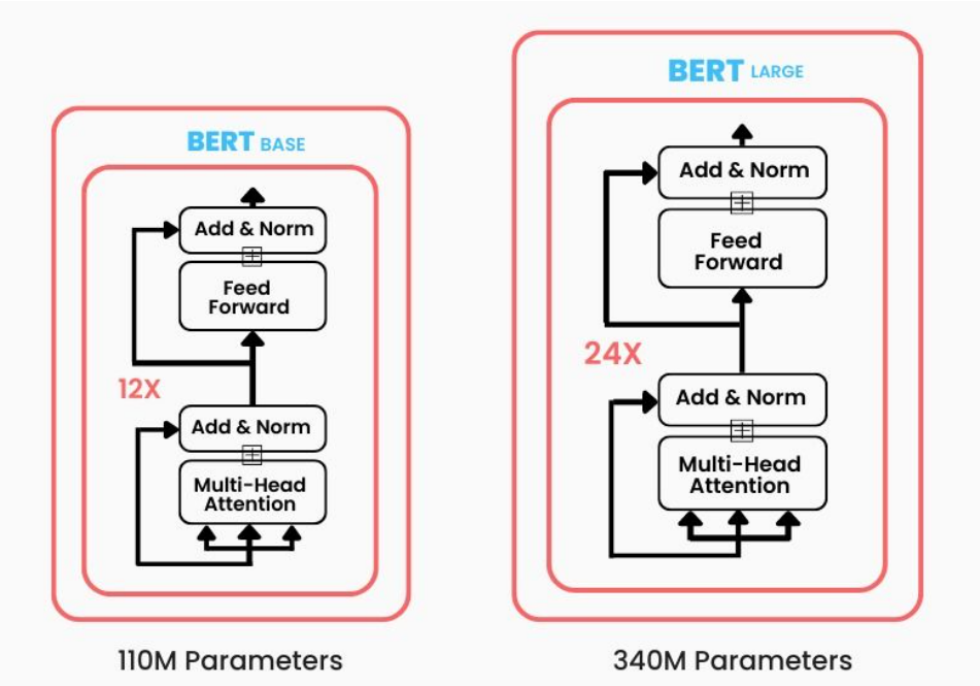


Figure 4.3: BERT Architecture

other. Figure 4.3 gives a high level view of the BERT architecture. Listed below are the components of the Encoder.

1. **Multi-Head Self-Attention Mechanism**: BERT employs a multi-head self-attention mechanism in its encoder layers. Self-attention allows each word in a sequence to attend to all other words, capturing dependencies between words regardless of their distance within the sequence. By using multiple attention heads, BERT can capture different aspects of inter-token relationships simultaneously.

2. **Position-wise Feed-Forward Networks**: After the self-attention mechanism, BERT applies position-wise feed-forward networks to each token independently. These feed-forward networks consist of multiple layers of fully connected neural networks, allowing BERT to model complex non-linear relationships between tokens.

4.1.2.3 Residual Connections and Layer Normalization

The output of the Encoder needs to be normalized before feeding it to the next encoder layer, since it plays a crucial role in maintaining the effectiveness and stability of BERT architecture.

1. **Residual Connections:** BERT uses residual connections, also known as skip connections, to connect the output of each layer to the input of the

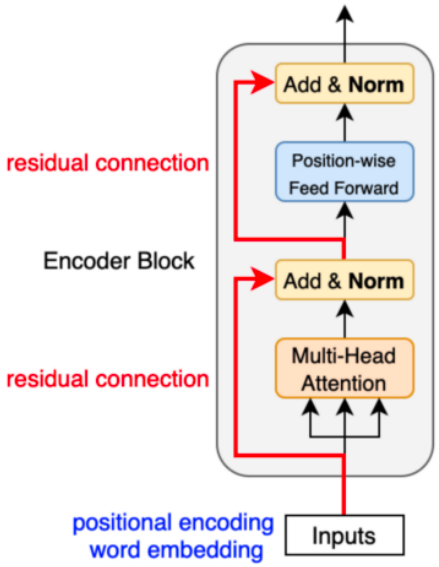


Figure 4.4: Residual Connections

subsequent layer. This helps in mitigating the vanishing gradient problem during training, enabling the model to effectively learn from deep architectures. Figure 4.4 shows examples of Residual Connections.

2. **Layer Normalization:** Layer normalization is applied after each sub-layer in the transformer encoder. It normalizes the activations of each layer, making the training process more stable and accelerating convergence. This normalization step contributes to BERT's ability to generalize well across different tasks and datasets.

4.2 Evaluation

Metrics for evaluation are crucial in determining a predictive/learning model's potency and usefulness. These metrics are used to assess how well a system performs in predicting the correct class label for an input in classification challenges. Considering a situation with two classes, where one class is designated as "positive" and the other as "negative."

- **True Positive** (T_P) refers to the number of accurately categorized positive samples.
- **False Positive** (F_P) refers to the number of negative samples that are mistakenly labeled as positive.
- **False Negative** (F_N) refers to those samples that are mistakenly labeled as negative despite being positive.

- **True Negative (T_N)** refers to the number of accurately categorized negative samples.

For the present work, the following performance metrics are used:

Precision: The proportion of accurate positive predictions amongst all positive predictions made by a model is known as its precision. It is determined by dividing T_P by the product of T_P and F_P. Mathematically, precision can be represented as in Equation 4.1.

$$Precision = \frac{T_P}{T_P + F_P} \tag{4.1}$$

Recall: Recall is the percentage of actual positive events that the model correctly identifies. It is the proportion of T_P to T_P plus F_N as shown in Equation 4.2. A higher recall demonstrates that the model is better at identifying positive examples, whereas a lower recall raises the possibility that some positive cases may have been missed by the model.

$$Recall = \frac{T_P}{T_P + F_N} \tag{4.2}$$

F1 Score: F1 Score is the harmonic mean of the two measures, Precision and Recall, ensuring that both false positives and false negatives are considered as shown in Equation 4.3. By combining these two, the F1 score provides a single measure that balances the trade-off between precision and recall, making it especially useful when the class distribution is imbalanced.

$$F1 - Score = \frac{2 * T_P}{(2 * T_P + F_N + F_P)} \tag{4.3}$$

4.3 Causality Extraction from CNC Dataset

We used a standard sequence tagging model that classifies each token in the sentence with BIO-style tags. The input text is tokenized with Huggingface's BertTokenizer [44]. For an input sequence, each token is assigned one of the following tags: {B-Cause, I-Cause, B-Effect, I-Effect, O}, where "B" stands for "Beginning", "I" for "Inside", and "O" for "Outside". Figure 4.6 shows such an example with the BIO tags.

We experiment with BERT-large [45] as the encoder for the input sentence. The BERT-large (bert-large-cased) model has 24 encoder layers with a token embedding dimension of 1024. We used the cased version of BERT-large since it preserves the original casing of entity names, helping the model differentiate between entities and other words in the text. This is particularly important when

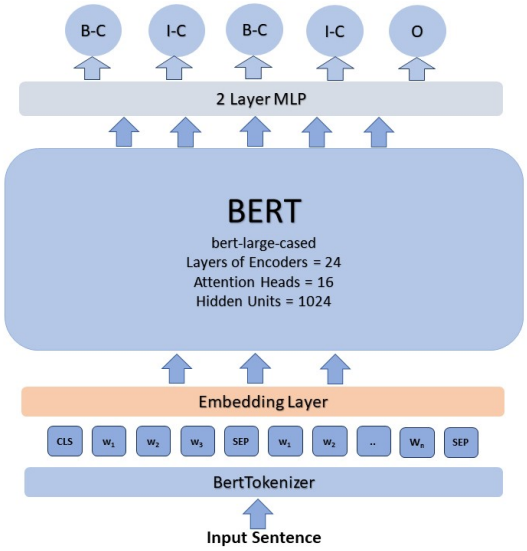


Figure 4.5: BERT for Token Classification Model Architecture

<ARG1>Four students appeared in court on Monday</ARG1> for <ARG0>allegedly removing street signs</ARG0> .

Four	students	appeared	in	court	on	Monday	for	allegedly	removing	street	signs	.
B-E	I-E	I-E	I-E	I-E	I-E	I-E	O	B-C	I-C	I-C	I-C	O

<ARG1>The workers had embarked on a wildcat strike</ARG1> <ARG0>demanding better working conditions</ARG0> .

The	workers	had	embarked	on	a	wildcat	strike	demanding	better	working	conditions	.
B-E	I-E	I-E	I-E	I-E	I-E	I-E	I-E	B-C	I-C	I-C	I-C	O

Figure 4.6: Examples showing input text labeled with ARG0 and ARG1 tags converted to BIO labels for Cause-and-Effect spans.

entity names are proper nouns or acronyms, where capitalization carries semantic significance.

The output from the BERT is the output[0] with dimensions (1, n, 1024), where 1 is the batch size, n is the sequence length, and each token has a corresponding hidden state vector of size 1024. The output[0] is chosen as it includes the hidden states of all tokens and provides token-level contextual information. The output contextual embedding (output[0]) from the BERT model is fed to a 2-layer MLP to classify each token into one of the 5 classes: [B-C, I-C, B-E, I-E, O] in the sentence. Figure 4.5 gives a high-level view of the model architecture.

4.3.1 Experimental Setup

We used the simpletransformers API [46] to implement our system. We use the train and development data during the model training phase. We fine-tuned our model setting the maximum sequence length to 16, training batch size and evaluation batch size to 8. The model was trained for 20 epochs. We configured the AdamW optimizer [47] with a learning rate of $4e^{-5}$.

4.3.2 Results

The results of the token classification model on the CNC dataset for the updated and the augmented versions are presented in Table 4.1. From the results obtained it can be seen that the Span Based Model by Saha et al. [39] performs similar to the Updated CNC version but since the datasets are different so the results are not comparable.

It can also be observed that the augmented version performs much better than the non-augmented version thus validating that augmenting to a dataset does add value since it introduces contextually relevant variations to the text.

	Dev Set			**Test Set**		
Dataset	**Precision**	**Recall**	**F1 Score**	**Precision**	**Recall**	**F1 Score**
CNC [39]	0.57	0.72	0.63	0.58	0.44	0.47
Updated CNC	0.55	0.71	0.62	0.57	0.41	0.46
Augmented CNC	0.593	0.715	0.65	0.502	0.627	0.56

Table 4.1: Results on the CNC Dataset

4.4 Causality Extraction from CNCB Dataset

We used a standard sequence tagging model, like the CNC dataset, that classifies each token in the sentence with BIO-style tags. The input text is tokenized with Huggingface's BertTokenizer [44]. For an input sequence, each token is assigned one of the following tags: {B-Cause, I-Cause, B-Effect, I-Effect, O}, where "B" stands for "Beginning", "I" for "Inside", and "O" for "Outside". Figure 4.8 shows such an example with the BIO tags in Bengali.

We experimented with the multilingual BERT (mBERT) [45] model as the encoder for the input sentence. The mBERT (bert-base-multilingual-cased) model has 12 encoder layers with a token embedding dimension of 768. We used the multilingual version of BERT-base since it allows for processing and understanding text in 104 languages including Bengali, which is the requirement in this case.

The output from the mBERT is the output[0] with dimensions (1, n, 768), where 1 is the batch size, n is the sequence length, and each token has a corresponding hidden state vector of size 768. The output[0] is chosen as it includes the hidden states of all tokens and provides token-level contextual information. The output contextual embedding (output[0]) from the mBERT model is fed to a 2-layer MLP to classify each token into one of the 5 classes: [B-C, I-C, B-E, I-E, O] in the sentence. Figure 4.7 gives a high-level view of the model architecture.

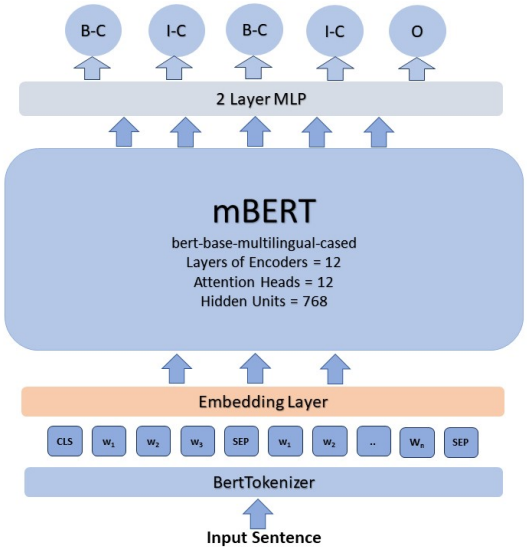


Figure 4.7: mBERT for Token Classification Model Architecture

<ARG0> গরু চুরির চেষ্টার অভিযোগে </ARG0><ARG1> গণপ্রহার </ARG1>। আর তাতে মৃত্যু হল দু'জনের।

গরু চুরির চেষ্টার অভিযোগে গণপ্রহার । আর তাতে মৃত্যু হল দু'জনের ।
B-C I-C I-C I-C B-E O O O O O O O

গরু চুরির চেষ্টার অভিযোগে <ARG0> গণপ্রহার </ARG0>। আর <ARG1> তাতে মৃত্যু হল দু'জনের </ARG1>।

গরু চুরির চেষ্টার অভিযোগে গণপ্রহার । আর তাতে মৃত্যু হল দু'জনের ।
O O O O B-C O O B-E I-E I-E I-E O

Figure 4.8: Examples showing input text labeled with ARG0 and ARG1 tags converted to BIO labels for Cause-and-Effect spans.

4.4.1 Experimental Setup

We used the simpletransformers API [46] to implement our system. We use the train and development data during the model training phase. We fine-tuned our model setting the maximum sequence length to 16, training batch size and evaluation batch size to 8. The model was trained for {10, 20, 30} epochs. We configured the AdamW optimizer [47] over a learning rate of $\{2e^{-5}, 4e^{-5}\}$.

4.4.2 Results

The result of the token classification model on the CNCB dataset is presented in Table 4.2. From the results obtained it can be observed that the test set result for 20 epochs for both the learning rates are similar and comparable. However, the F1-score for the validation set differs by 0.052.

Parameters		**Validation Set**			**Test Set**		
No of Epochs	L.R	Precision	Recall	F1 Score	Precision	Recall	F1 Score
10	$2e^{-5}$	0.7	0.875	0.77	0.63	0.875	0.736
20		0.77	0.875	**0.823**	0.77	0.875	**0.823**
30		0.77	0.875	0.823	0.7	0.875	0.77
10	$4e^{-5}$	0.7	0.87	0.77	0.5	0.75	0.6
20		0.875	0.875	**0.875**	0.7	0.875	**0.823**
30		0.77	0.875	0.823	0.77	0.875	0.823

Table 4.2: Results on the CNCB Dataset

4.5 Causality Extraction from CNCB-QA Dataset

We experiment with the CNCB-QA dataset to extract the causes and effects from sentences through question answering. For that we divided the task into two subtasks – the cause extraction subtask and the effect extraction subtasks which are described below:

- **Cause Extraction Sub-task** – This sub-task aims at extracting the cause of an event by framing the effect as a question and the cause is an answer to the question. For Multiple Cause One Effect (MCOE) category, multiple answers (causes) are possible for a single question (effect).
- **Effect Extraction Sub-task** – This sub-task aims at extracting the effect of an event by framing the cause as a question and the effect is an answer to the question. For One Cause Multiple Effect (OCME) category, multiple answers (effects) are possible for a single question (cause).

We used the BERT for QA model which takes as input the question and the context. The input is then tokenized with Huggingface's BertTokenizer [44]. We experimented with the multilingual BERT (mBERT) [45] model as the encoder for the input. The mBERT (bert-base-multilingual-cased) model has 12 encoder layers with a token embedding dimension of 768. We used the multilingual version of BERT-base since it allows for processing and understanding text in 104 languages including Bengali, which is the requirement in this case.

Unlike the BERT for token classification model which takes output[0], we take the output[1] with dimension (1,768), where 1 is the batch size and each token has a corresponding hidden state vector of size 768. For question answering tasks, the pooled output (outputs[1]) is often used as it provides a high-level representation of the entire input sequence, which is beneficial for understanding the overall context and locating the answer span. The pooled output contextual embedding

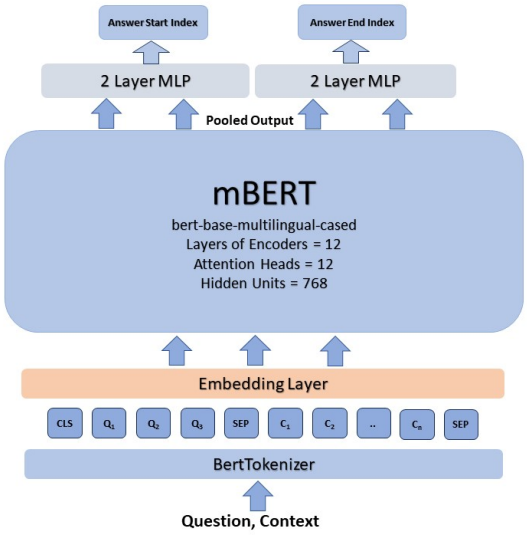


Figure 4.9: mBERT for Question Answering Model Architecture

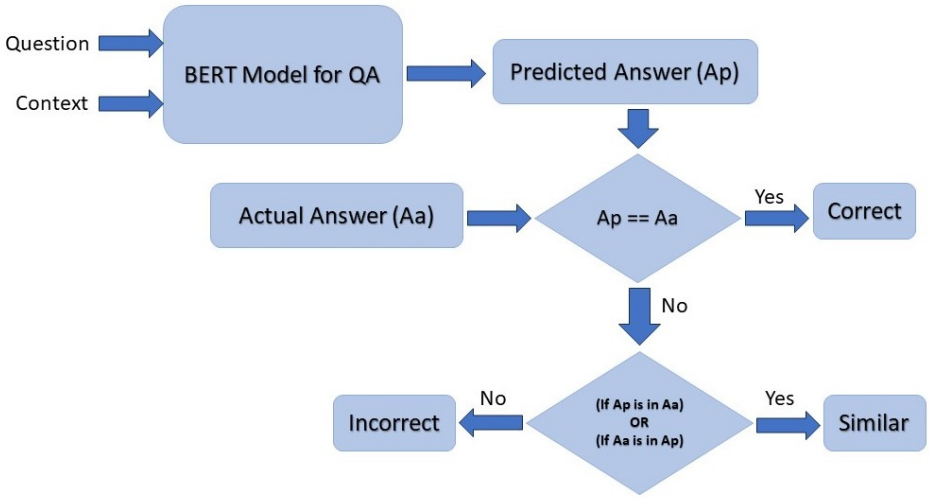


Figure 4.10: Flowchart for Predicted Answer Classification

(output[1]) from the mBERT model is fed to two different 2-layer MLP that gives as output the answer starting index and the answer ending index. Figure 4.9 gives a high-level view of the model architecture.

The starting and ending indices are used to locate and extract the predicted answer from the context. Once the predicted answer is extracted it is compared with the actual answer and if it is an exact match then it is classified as correct. If the actual answer is a substring of the predicted answer or the predicted answer is a substring of the actual answer then it is classified as similar else incorrect. This process is illustrated through a flowchart in Figure 4.10.

4.5.1 Experimental Setup

We used the simpletransformers API [46] to implement our system. We use the train and development data during the model training phase. We fine-tuned our model setting the maximum sequence length to 400, training batch size and evaluation batch size to 8. The model was trained for 10 epochs. We configured the AdamW optimizer [47] with a learning rate of $4e^{-5}$.

The training was done separately for the cause extraction and effect extraction subtasks.

4.5.2 Results

The result of the question answering model on the CNCB-QA dataset is presented in Table 4.3. From the results obtained it can be observed that we achieved a 100% similarity score for both validation and test dataset for both cause and event extraction subtask.

Sub-Task	**Validation Set**			**Test Set**		
	Correct	**Similar**	**Incorrect**	**Correct**	**Similar**	**Incorrect**
Effect Extraction	0	17	0	0	15	0
Cause Extraction	0	16	0	0	12	0

Table 4.3: Results on the CNCB-QA dataset

CHAPTER 5

Conclusions

This thesis work tries to understand and study the nature of events and how they are temporally and causally related to each other. Cause and Effect relationships or Causality between events has been studied philosophically at great length however not much studying has been done for practical applications given that it is very common to occur almost in every possible field of study such as biomedical, legal, financial, socio-political etc. In recent years, the study of causality has gained great attention, particularly in practical realms, propelled by advancements in Natural Language Processing (NLP).

5.1 Conclusions of the Present Work

The study aims at studying event causality majorly in the socio-political realm. The Causal News Corpus (CNC) dataset is in English that aims at listing social events from all around the world thus studying, understanding and extracting the causes and effects from them. However, while studying the CNC dataset, we understood the need and importance for the existence of such dataset in Indian Languages as well. Hence, we created and annotated the Causal News Corpus Bengali (CNCB) dataset which lists socio-political events across the world annotated with cause, effect spans in Bengali. The Causal News Corpus Bengali Question Answering (CNCB-QA) dataset is derived from the CNCB dataset, also in Bengali which tries to extract the cause-and-effect spans through question answering. The dataset contains 100 articles collected carefully that covers events from around the world.

We experimented with the CNC dataset by augmenting it with new data generated from the LLaMa 2 LLM. The F1-Score for the test data increased by 10% by augmenting only 56 sentences having 103 cause-effect relationships to the training data. The CNCB dataset gave an impressive 82.3% F1-Score for

cause-effect extraction through token classification and the CNCB-QA dataset gave 100% similarity score for both event and cause extraction subtasks through question answering.

5.2 Scope for Future Works

Causality studies in events is a field that is experiencing rapid growth and in the future such studies need to be expanded into other domains such as biomedical, legal and financial. In addition to this studies also need to be conducted in other languages including Indian languages since in India legal and governance systems operate in multiple languages. Also, Indian society is characterized by its diversity, and social and political discussions often take place in regional languages.

In the future, expanding the CNCB dataset and in turn the CNCB-QA dataset will be crucial as it will allow for the inclusion of newer and significant events. This expansion will contribute to a deeper understanding of causality studies by broadening the scope of captured phenomena. In addition to that new datasets in other Indian languages in domains like finance, law and policy making might have a great impact in understanding causality of events that directly impact our society.

Bibliography

[1] ACE, "English annotation guidelines for events," *Linguistic Data Consortium, Philadelphia, PA, USA*, 2005.

[2] W. Xiang and B. Wang, "A survey of event extraction from text," *IEEE*, vol. 7, Dec. 2019.

[3] P.Mirza, "Extracting temporal and causal relations between events," *Association for Computational Linguistics*, vol. 7, pp. 10–17, Jun. 2014.

[4] F. A. Tan, A. Hürriyetoğlu, T. Caselli, *et al.*, "The causal news corpus: Annotating causal relations in event sentences from news," in *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, N. Calzolari, F. Béchet, P. Blache, *et al.*, Eds., Marseille, France: European Language Resources Association, Jun. 2022, pp. 2298–2310.

[5] W. Ali, W. Zuo, W.Ying, R.Ali, G. Rahman, and I.Ullah, "Causality extraction: A comprehensive survey and new perspective," *Elsevier*, 2023.

[6] Garcia.D, EDF-DER, and IMA-TIEM, "Coatis, an nlp system to locate expressions of actions connected by causality links," in *Springer*, ser. chap BFb0026799, vol. 1319, 2006, pp. 347–352. DOI: 10.1007/BFb0026799.

[7] Radinsky.K, Davidovich.S, and Markovitch.S, "Learning causality for news events prediction," in *WWW'12—Proceedings of the 21st Annual Conference on World Wide Web*, 2012, pp. 909–918. DOI: 10.1145/2187836.2187958.

[8] Ittoo.A and Bouma.G, "Minimally-supervised learning of domain-specific causal relations using an open-domain corpus as knowledge base," *Data & Knowledge Engineering*, vol. 88, pp. 142–163, 2013. DOI: 10.1016/j.datak.2013.08.004.

[9] Kang.N, Singh.B, Bui.QC, Afzal.Z, van Mulligen.EM, and Kors.J, "Knowledge-based extraction of adverse drug events from biomedical text," *BMC Bioinformatics*, vol. 15, p. 64, 2014. DOI: 10.1186/1471-2105-15-64.

[10] Khoo.CSG, Kornfilt.J, Oddy.RN, and Myaeng.SH, "Automatic extraction of cause-effect information from newspaper text without knowledge-based inferencing," *Literary and Linguistic Computing*, vol. 13, no. 4, pp. 177–186, 1998. DOI: 10.1093/llc/13.4.177.

[11] Girju.R, "Automatic detection of causal relations for question answering," in *Proceedings of the ACL 2003 Workshop on Multilingual Summarization and Question Answering (MultiSumQA'03)*, Association for Computational Linguistics, vol. 12, USA, 2003, pp. 76–83. DOI: 10.3115/1119312.1119322.

[12] Blanco.E, Castell.N, and Moldovan.D, "Causal relation extraction," in *Proceedings of the International Conference on Language Resources and Evaluation*, Marrakech, Morocco, 2008, pp. 310–313.

[13] Pakray.P and Gelbukh.A, "An open domain causal relation detection from paired nominal," in *13th Mexican International Conference on Artificial Intelligence (MICAI-2014). Nature-Inspired Computation and Machine Learning*, vol. 8857, Chiapas, Mexico, 2014, pp. 261–271. DOI: 10.1007/978-3-319-13650-9_24.

[14] Sorgente.A, Vettigli.G, and Mele.F, "Automatic extraction of cause-effect relations in natural language text," in *DART@AI*IA 2013*, vol. 1109, 2013, pp. 37–48.

[15] Zhao.S, Liu.T, Zhao.S, Chen.Y, and NieJ.Y, "Event causality extraction based on connectives analysis," *Neurocomputing*, vol. 173, pp. 1943–1950, 2016. DOI: 10.1016/j.neucom.2015.09.066.

[16] Hidey.C and McKeown.K, "Identifying causal relations using parallel wikipedia articles," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, Berlin, Germany, 2016, pp. 1424–1433. DOI: 10.18653/v1/P16-1135.

[17] Airola.A, Pyysalo.S, Björne.J, Pahikkala.T, Ginter.F, and Salakoski.T, "All-paths graph kernel for protein-protein interaction extraction with evaluation of cross-corpus learning," *BMC Bioinformatics*, vol. 9, no. 11, S2, 2008. DOI: 10.1186/1471-2105-9-S11-S2.

[18] Keskes.I, Zitoune.FB, and Belguith.L, "Learning explicit and implicit arabic discourse relations," *Journal of King Saud University - Computer and Information Sciences*, vol. 26, pp. 398–416, 2014. DOI: 10.1016/j.jksuci.2014.06.001.

[19] Pechsiri.C, Kawtrakul.A, and Piriyakul.R, "Mining causality knowledge from textual data," in *Proceedings of the 24th IASTED International Conference on Artificial Intelligence and Applications*, ACTA Press, USA, 2006, pp. 85–90.

[20] Gurulingappa.H, Rajput.AM, Roberts.A, Fluck.J, Hofmann-Apitius.M, and Toldo.L, "Development of a benchmark corpus to support the automatic extraction of drug-related adverse effects from medical case reports," *Journal of Biomedical Informatics*, vol. 45, no. 5, pp. 885–892, 2012. DOI: 10.1016/j.jbi.2012.04.008.

[21] Marcu.D and Echihabi.A, "An unsupervised approach to recognizing discourse relations," in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Philadelphia, Pennsylvania, USA, 2002, pp. 368–375. DOI: 10.3115/1073083.1073145.

[22] Oh.JH, Torisawa.K, Hashimoto.C, Sano.M, D. Saeger.S, and Ohtake.K, "Why-question answering using intra- and inter-sentential causal relations," in *ACL 2013—51st Annual Meeting of the Association for Computational Linguistics. Proceedings of the Conference*, vol. 1, Sofia, Bulgaria, 2013, pp. 1733–1743.

[23] Qian.L and Zhou.G, "Chemical-induced disease relation extraction with various linguistic features," *Database*, vol. 2016, baw042, 2016. DOI: 10.1093/database/baw042.

[24] J. Yang, S. C. Han, and J. Poon, "A survey on extraction of causal relations from natural language text," *Knowledge and Information Systems*, vol. 64, no. 5, pp. 1161–1186, 2022, ISSN: 0219-3116. DOI: 10.1007/s10115-022-01665-w.

[25] Xu.Y, Mou.L, Li.G, Chen.Y, Peng.H, and Jin.Z, "Classifying relations via long short term memory networks along shortest dependency paths," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Lisbon, Portugal, 2015, pp. 1785–1794. DOI: 10.18653/v1/D15-1206.

[26] Li.Z, Li.Q, Zou.X, and Ren.J, "Causality extraction based on self-attentive bilstm-crf with transferred embeddings," *Neurocomputing*, vol. 423, pp. 207–219, 2021. DOI: 10.1016/j.neucom.2020.08.078.

[27] Kyriakakis.M, Androutsopoulos.I, Saudabayev.A, and G. A. J, "Transfer learning for causal sentence detection," in *Proceedings of the 18th BioNLP Workshop and Shared Task*, Association for Computational Linguistics, Florence, Italy, 2019, pp. 292–297. DOI: 10.18653/v1/W19-5031.

[28] Martínez-Cámara.E, Shwartz.V, Gurevych.I, and Dagan.I, "Neural disambiguation of causal lexical markers based on context," in *IWCS 2017—12th International Conference on Computational Semantics—Short Papers*, 2017.

[29] Li.F, Zhang.M, Fu.G, and Ji.D, "A neural joint model for entity and relation extraction from biomedical text," *BMC Bioinformatics*, vol. 18, 2017. DOI: 10.1186/s12859-017-1609-9.

[30] J. Wang and W. Lu, "Two are better than one: Joint entity and relation extraction with table-sequence encoders," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, B. Webber, T. Cohn, Y. He, and Y. Liu, Eds., Online: Association for Computational Linguistics, Nov. 2020, pp. 1706–1721. DOI: 10.18653/v1/2020.emnlp-main.133.

[31] Zhao.S, Hu.M, Cai.Z, and Liu.F, "Modeling dense cross-modal interactions for joint entity-relation extraction," in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, C. Bessiere, Ed., International Joint Conferences on Artificial Intelligence Organization, 2020, pp. 4032–4038. DOI: 10.24963/ijcai.2020/558.

[32] Dasgupta.T, Saha.R, Dey.L, and Naskar.A, "Automatic extraction of causal relations from text using linguistically informed deep neural networks," in *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue*, Association for Computational Linguistics, Melbourne, Australia, 2018, pp. 306–316. DOI: 10.18653/v1/W18-5035.

[33] J. Kim, Y. Choe, and S. Lee, "SNU-causality lab @ causal news corpus 2022: Detecting causality by data augmentation via part-of-speech tagging," in *Proceedings of the 5th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text (CASE)*, A. Hürriyetoğlu, H. Tanev, V. Zavarella, and E. Yörük, Eds., Abu Dhabi, United Arab Emirates (Hybrid): Association for Computational Linguistics, Dec. 2022, pp. 44–49. DOI: 10.18653/v1/2022.case-1.6.

[34] R. Raj, K. Andreasson, T. Norlund, R. Johansson, and A. Lagerberg, "Cross-modal transfer between vision and language for protest detection," in *Proceedings of the 5th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text (CASE)*, A. Hürriyetoğlu, H.

Tanev, V. Zavarella, and E. Yörük, Eds., Abu Dhabi, United Arab Emirates (Hybrid): Association for Computational Linguistics, Dec. 2022, pp. 56–60. DOI: 10.18653/v1/2022.case-1.8.

[35] M. Fajcik, M. Singh, J. P. Zuluaga-gomez, *et al.*, "IDIAPers @ causal news corpus 2022: Extracting cause-effect-signal triplets via pre-trained autoregressive language model," in *Proceedings of the 5th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text (CASE)*, A. Hürriyetoğlu, H. Tanev, V. Zavarella, and E. Yörük, Eds., Abu Dhabi, United Arab Emirates (Hybrid): Association for Computational Linguistics, Dec. 2022, pp. 70–78. DOI: 10.18653/v1/2022.case-1.10.

[36] P. Trust, R. Minghim, E. Milos, and K. Provia, "GGNN@causal news corpus 2022:gated graph neural networks for causal event classification from social-political news articles," in *Proceedings of the 5th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text (CASE)*, A. Hürriyetoğlu, H. Tanev, V. Zavarella, and E. Yörük, Eds., Abu Dhabi, United Arab Emirates (Hybrid): Association for Computational Linguistics, Dec. 2022, pp. 85–90. DOI: 10.18653/v1/2022.case-1.12.

[37] X. Chen, G. Zhang, A. Nik, M. Li, and J. Fu, "1Cademy @ causal news corpus 2022: Enhance causal span detection via beam-search-based position selector," in *Proceedings of the 5th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text (CASE)*, A. Hürriyetoğlu, H. Tanev, V. Zavarella, and E. Yörük, Eds., Abu Dhabi, United Arab Emirates (Hybrid): Association for Computational Linguistics, Dec. 2022, pp. 100–105. DOI: 10.18653/v1/2022.case-1.14.

[38] K. Akdemir and A. Hürriyetoğlu, "Zero-shot ranking socio-political texts with transformer language models to reduce close reading time," in *Proceedings of the 5th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text (CASE)*, A. Hürriyetoğlu, H. Tanev, V. Zavarella, and E. Yörük, Eds., Abu Dhabi, United Arab Emirates (Hybrid): Association for Computational Linguistics, Dec. 2022, pp. 124–132. DOI: 10.18653/v1/2022.case-1.17.

[39] A. Saha, A. Gittens, J. Ni, O. Hassanzadeh, B. Yener, and K. Srinivas, "SPOCK @ causal news corpus 2022: Cause-effect-signal span detection using span-based and sequence tagging models," in *Proceedings of the 5th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text (CASE)*, A. Hürriyetoğlu, H. Tanev, V. Zavarella, and E. Yörük, Eds., Abu Dhabi, United Arab Emirates (Hybrid): Associa-

tion for Computational Linguistics, Dec. 2022, pp. 133–137. DOI: 10.18653/v1/2022.case-1.18.

[40] F. A. Tan, H. Hettiarachchi, A. Hürriyetoğlu, *et al.*, "RECESS: Resource for extracting cause, effect, and signal spans," in *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, J. C. Park, Y. Arase, B. Hu, *et al.*, Eds., Nusa Dua, Bali: Association for Computational Linguistics, Nov. 2023, pp. 66–82. DOI: 10.18653/v1/2023.ijcnlp-main.6.

[41] A. Hürriyetoğlu, V. Zavarella, H. Tanev, E. Yörük, A. Safaya, and O. Mutlu, "Automated extraction of socio-political events from news (AESPEN): Workshop and shared task report," English, in *Proceedings of the Workshop on Automated Extraction of Socio-political Events from News 2020*, A. Hürriyetoğlu, E. Yörük, V. Zavarella, and H. Tanev, Eds., Marseille, France: European Language Resources Association (ELRA), May 2020, pp. 1–6, ISBN: 979-10-95546-50-4.

[42] A. Hürriyetoğlu, H. Tanev, V. Zavarella, *et al.*, "Challenges and applications of automated extraction of socio-political events from text (CASE 2021): Workshop and shared task report," in *Proceedings of the 4th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text (CASE 2021)*, A. Hürriyetoğlu, Ed., Online: Association for Computational Linguistics, Aug. 2021, pp. 1–9. DOI: 10.18653/v1/2021.case-1.1.

[43] H. Touvron, T. Lavril, G. Izacard, *et al.*, "LLaMA: Open and Efficient Foundation Language Models," 2023.

[44] A. Moi and N. Patry, *Huggingface's tokenizers*, version 0.13.4, 2023.

[45] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "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)*, J. Burstein, C. Doran, and T. Solorio, Eds., Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186. DOI: 10.18653/v1/N19-1423.

[46] T. C. Rajapakse, *Simple transformers*, https://github.com/ThilinaRajapakse/simpletransformers, 2019.

[47] I. Loshchilov and F. Hutter, *Decoupled weight decay regularization*, 2019.