TU WIEN Informatics

# Advanced pattern matching in graph-based relation extraction

## A methodical approach to improving XAI NLP systems

DIPLOMARBEIT

zur Erlangung des akademischen Grades

**Diplom-Ingenieur**

im Rahmen des Studiums

**Data Science**

eingereicht von

**Philipp Piwonka, BSc**
Matrikelnummer 12002294

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: Univ.Ass. Gábor Recski, PhD
Mitwirkung: Ádám Kovács, MSc



Wien, 27. August 2024

| Signed by: | Gabor Andras Recski |
|---|---|
| Date: | 27.08.2024 13:04:15 |

This document is digitally signed!
This document bearing a qualified electronic signature has the same legal standing as a handwritten document in accordance with Article 25 (2) of Regulation (EU) No. 910/2014 of 23 July 2014 ("eIDAS-VO").
Validation: Information for checking the electronic signature can be found at: www.a-trust.at/pdf

Philipp Piwonka

Gábor Recski

TU WIEN Informatics

# Advanced pattern matching in graph-based relation extraction

## A methodical approach to improving XAI NLP systems

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

**Diplom-Ingenieur**

in

**Data Science**

by

**Philipp Piwonka, BSc**
Registration Number 12002294

to the Faculty of Informatics

at the TU Wien

Advisor: Univ.Ass. Gábor Recski, PhD
Assistance: Ádám Kovács, MSc

Vienna, August 27, 2024

Philipp Piwonka | Gábor Recski

Technische Universität Wien
A-1040 Wien ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.at

# Erklärung zur Verfassung der Arbeit

Philipp Piwonka, BSc

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.
Ich erkläre weiters, dass ich mich generativer KI-Tools lediglich als Hilfsmittel bedient habe und in der vorliegenden Arbeit mein gestalterischer Einfluss überwiegt. Im Anhang „Übersicht verwendeter Hilfsmittel“ habe ich alle generativen KI-Tools gelistet, die verwendet wurden, und angegeben, wo und wie sie verwendet wurden. Für Textpassagen, die ohne substantielle Änderungen übernommen wurden, haben ich jeweils die von mir formulierten Eingaben (Prompts) und die verwendete IT- Anwendung mit ihrem Produktnamen und Versionsnummer/Datum angegeben.

Wien, 27. August 2024

Philipp Piwonka

# Acknowledgements

I want to thank my advisor, Univ.Ass. Gábor Recski, PhD, for his continuous support on this thesis. Not only has he been a constant source of valuable feedback, inspiration and insight, but his infectious passion for NLP has given me the motivation to see this thesis through.
I also want to thank the staff and professors at TU Vienna, who have been supportive of my efforts during my years of study all the way to this final stretch.
And finally, my parents, for always being there for me.

# Kurzfassung

Die Identifizierung und Klassifizierung von semantischen Beziehungen zwischen Entitäten eines gegebenen Textes ist ein fortbestehendes Problem der wissenschaftlichen Arbeit im Bereich des Natural Language Processings (NLP). In der Praxis sind es oft die transformer- oder neural-network-basierten Modelle, die weitgehend die besten Performance-Metriken aufweisen. Diese Black-Box-Modelle sind aber oft schwer bis unmöglich zu interpretieren, und damit für sprachwissenschaftliche Experimente wenig geeignet. Eine Alternative dazu ist POTATO, ein Framework, das sich auf den Bau von erklärbaren Text-Klassifizerungs-Modellen fokusiert. Es repräsentiert Text in mehreren etablierten syntaktischen und semantischen Graph-Systemen, und erlaubt es einem Menschen, durch einen iterativen Prozess graduell ein erklärbares Klassifizierungs-Modell aufzubauen, das transparente Entscheidungen auf Basis einer Pattern-Matching-Logik trifft. In dieser Diplomarbeit bauen wir einen solches Modell zur Klassifizierung des CrowdTruth Cause Datasets, einem binären Klassifizierungs-Problem für Entity Relations. Durch diesen Prozess wollen wir systematisch Verbesserungspotenzial an POTATO untersuchen und konkrete Verbesserungsvorschläge identifizieren und ausformulieren. In einem weiteren Schritt setzen wir eine unserer vorgeschlagenen Maßnahmen um: anstatt Entities durch Platzhalter-Text zu ersetzen, um sie so im konvertierten Graphen erkennen zu können, führen wir ein System ein, das Entity-Nodes in Graphen per Attribut markiert und dabei den Ursprungs-Text der Entity erhält. Wir demonstrieren, wie dieses neue Tagging-System zu einem besseren Klassifizierungs-Ergebnis führt, da es zum einen die Leistung existierender Regeln erhöht, und zum anderen die Erstellung ganz neuer Graph-Patterns ermöglicht, bei denen direkt die Inhalte von Entity-Nodes zur Klassifierung verwendet werden können. Dies führt zu einer Steigerung der F1-Test-Score, von 0,31 im Ursprungssystem zu 0,35 mit neuem Tagging-System und Graph-Patterns die den Inhalt von Entity-Nodes referenzieren.

# Abstract

The identification and classification of relations between entities is an ongoing concern of research on natural language processing (NLP). In practice, transformer-based models provide state-of-the-art performance on such tasks, but they are borderline impossible to interpret and therefore ill-suited for linguistic research purposes. POTATO is an alternative that provides a framework for building explainable text classification models through an iterative process with a human agent, using established graph representation systems for natural language and a transparent pattern-matching logic. In this thesis we build a ruleset on the CrowdTruth Cause dataset, a binary entity relation classification problem, to systematically explore, identify and propose opportunities for improvement of the POTATO framework. In a further step, we take up the implementation of one such proposal: instead of marking entities by replacing their text with placeholder strings, we demonstrate an entity tagging mechanic that preserves the original text in entity nodes of syntactic and semantic graphs. We demonstrate how this new mechanic can be beneficial to the process of creating an explainable ruleset, as it enhances the performance of existing rules, and enables the building of entirely new types of patterns that specifically target entity node labels for classification. This leads to an overall improvement in performance metrics on the classification task, from a 0.31 F1 test metric on the original classifier, to an F1 of 0.35 when using the new tagging system and entity-content-aware rules.

# Contents

CHAPTER 1

# Introduction

Named entity recognition (NER) and relation extraction (RE) are ongoing problems in natural language processing (NLP) studies. While there are well-performing systems for many benchmark tasks, the most effective predictors are often based on black-box models built with transformer- or other types of neural network architecture. This comes at the cost of interpretability and reliability of a given classification system, which is a concern especially when we want to use it for NLP research.

POTATO is a framework for building explainable text classification systems using a human-in-the-loop approach as its core design principle. It can map text to a number of graph representation systems from NLP research to encode natural language in a variety of ways. These graphs can be understood and modified by a human agent, who can define subgraph patterns and pass them on to POTATO's built-in pattern matcher. This allows them to iteratively build a set of rules when investigating a given classification problem. Additionally, the performance of any model built with POTATO can be analyzed and debugged to a very fine level, since it is trivial to identify which pattern matched with which observation, and why.

In this thesis we systematically build an entity relation classification system to explore the current state of POTATO's implementation and identify opportunities for improvement. We then explain and demonstrate these opportunities. Amongst others, we will find that the current conversion algorithm cannot mark entity nodes in graphs non-destructively. Instead the reference workflow uses a preprocessing step wherein entities are replaced by placeholder strings for later identification. We propose a non-destructive alternative that marks entities by assigning attributes to the respective nodes at the point of graph conversion. We then demonstrate that our new tagging system leads to an improvement in precision and recall scores on existing rulesets, and enables the creation of a whole new class of rules that incorporates the contents of entity nodes for classification. All in all the new tagging system and the addition of only a few entity-content-aware rules allows us to boost the test F1 score of our classifier from 0.31 to 0.35.

In **Chapter 2**, this thesis will begin by establishing a theoretical background for relation extraction problems, state-of-the-art solutions, and the POTATO framework's advantages over technically more accurate classifiers. It will then give an overview of the POTATO workflow, its current feature set, and some of its technical limitations. We use the theoretical foundations outlined in this chapter as a basis to present our research questions for the thesis.

In **Chapter 3** we explain the methodology by which we conduct our experiments, as well as a list of proposed changes to the POTATO framework itself. These suggestions are based on our own work with the POTATO system and include explanations and examples for a range of possible improvements, such as additions to the rule-building syntax, or a non-destructive conversion system for entity relation problems.

In **Chapter 4** we document our experiments with the POTATO framework. First we investigate the current capabilities of POTATO by building a new classification model for an entity relation classification task. This process allows us to illustrate potential complications with the framework by building a new ruleset and documenting pitfalls as they occur. These findings serve as direct foundation for many of the suggestions we outline in chapter 3.

In a further step, we document experiments with a new entity tagging system we implement for this thesis. We will again provide precise examples for how the new tagging system affects the structure of existing graphs and the performance of the existing rule system. We also demonstrate an approximation of how the non-destructive entity tagging system could be used to incorporate domain knowledge into a classifier model by referencing the contents of entity nodes to build previously impossible and predictive rules.

**Chapter 5** will examine the results of our experiments. In particular, we evaluate the performance of our custom-made rulesets and conclude on the usefulness of our new entity tagging system. We also discuss ideas for future expansion and improvements to this new feature.

Finally, **Appendices I and II** contain all notes and the full ruleset built in the initial experiment. **Appendix III** contains the entire ruleset following the implementation of the new tagging system. Metrics for all individual rules can also be found in these appendices.

Our contributions to POTATO's entity marking system are available for public use as open-source-software in forks from the POTATO[1] and TUW-NLP[2] repositories respectively.

Unless otherwise stated, pictures in this thesis have been generated using POTATO's implementation of networkx[3] and graphviz[4] libraries.

[1] https://github.com/Entenzahn/POTATO-relation-entities/tree/entity_marking
[2] https://github.com/Entenzahn/tuw-nlp-relation-entities/tree/entity_marking
[3] https://pypi.org/project/networkx/
[4] https://pypi.org/project/graphviz/

CHAPTER 2

# Background

## 2.1 Natural Language Processing

Natural Language Processing (NLP) is a discipline of Data Science that strongly intersects with linguistic research. Machines have a much different understanding of language than humans – a word by itself is just another unit of data, perhaps distinct from other words, but semantically meaningless to an algorithm. NLP provides methods, research and formalisms that allow us to represent language in machine-readable formats, preserve its relevant features for a given task and further use those representations to solve problems through the use of machine learning or statistical methods. Implicitly, to encode any kind of meaning into a format that can be read and processed sensibly by a machine, human knowledge will be involved, for example represented through a large pre-trained language model, or simply by a human agent that interacts with a classification system.

Common problems of NLP include:

- Classification: detect spam, sentiment analysis
- Multiclass: detect keywords, detect genre
- Generative: summarize text, respond to prompts

A subset of text classification problems is relation extraction. A standard classification machine might be given a sentence, "Aspirin eliminates headaches" (a well-known example in relation inference research (Levy & Dagan, 2016)), and be asked to determine if there is a medical context to it. On the contrary, a relation classifier will be given additional entity markers $E_1 ... E_n$. These markers correspond to specific positions in the input. For instance, we might be given the sentence "Aspirin eliminates headaches" again, but this time with the words $E_1$ = "*Aspirin*" and $E_2$ = "*headaches*" marked as entities. The

system could then be asked to determine if there is a medical treatment relationship between these two entities.

## 2.2 Related Work

Like most classification problems, there is a long history of using machine learning methods to solve NLP tasks. For text classification in particular, there are multiple other considerations, such as how to represent and encode the words.

Before we get into the theory, it is important to clear up some common concepts for later reference:

- Tokenization: A given text is split up into separate tokens according to some logic. Generally, this is used to separate text into a list of words and punctuation, so that they can be further processed as individual units.
- Part-of-speech (POS) tagging: this process assigns various types of grammatical categorization to each word in a given piece of text. The type of attributes that can be assigned depends on the tagging system itself, as well as the type of the word. For example, a noun might have a case or gender, while a word that is tagged as a verb could additionally be tagged with a tense.
- Lemmatization: In a further step, tokens are often reduced to their lemmas, stripping conjunctions, inflections and other grammatical modifications to represent a given word in its grammatical base form.

The above three methods are bread-and-butter techniques of preprocessing in many NLP problems. An illustration of the POS-tagging and lemmatization process on a tokenized sentence can be seen in figure 2.1.

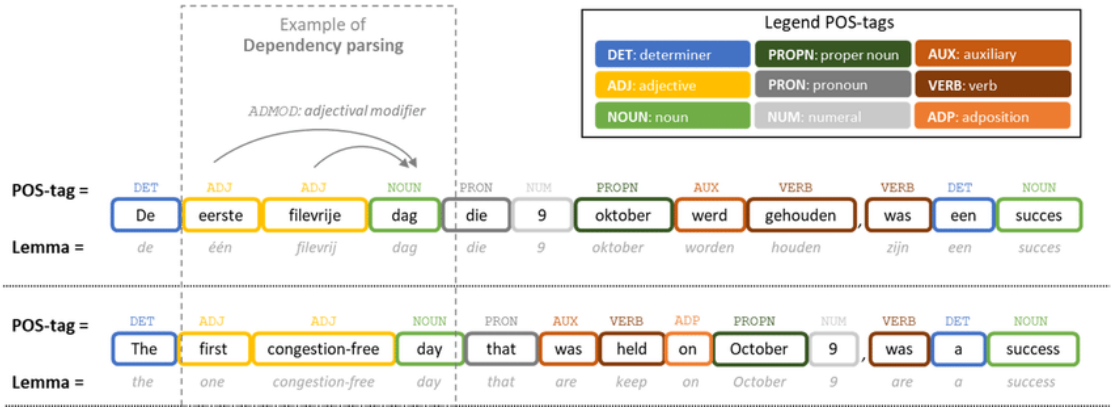



Figure 2.1: POS-tagging and lemmatization of a tokenized sentence in Dutch and English. Source: Manders and Klaassen (2019)

Further terms of relevance:

- Treebank: A treebank is a corpus of text with fully POS-tagged entities and relations between them often modeled in a tree-like structure. These treebanks represent important gold standards in understanding a given language and can be used for the analysis and training of models.

- Transformers (Vaswani et al., 2023): Transformers are architectural units used in the building of modern neural networks for dealing with language. The main benefit of a transformer is that it uses a concept called 'Self-attention' which allows it to consider the context of a given word when computing its encoding.

- Vector representation: one way to represent language in a machine-readable format is to encode it as numerical vectors. There is a number of approaches for this. For instance, GloVe (Pennington, Socher, & Manning, 2014) establishes a word-to-vector dictionary while transformer-based models such as BERT (Devlin, Chang, Lee, & Toutanova, 2019) encode word tokens into vectors by taking their surrounding context into account. These vector encodings can then be used for further downstream tasks, such as similarity analysis, translation or question-answering.

- Large language model: a transformer-based model that has been trained on extensive amounts of data to produce output from a given input. Commonly used to generate human speech. (Minaee et al., 2024)

As for the contemporary state-of-the-art, for many language classification tasks, the best-performing models are based on transformer-type architecture. These models consist of sprawling neural networks, sometimes with up to billions of parameters and trained on extensive amounts of data sourced from the internet. (Fields, Chovanec, & Madiraju, 2024)

However, there is a number of problems with these types of models. From an economic standpoint, these enormous models are costly to train (Bender, Gebru, McMillan-Major, & Shmitchell, 2021). From a legal standpoint, the question of copyright in training data for large AI models is an ongoing concern and subject of research (Wang, Deng, Chiba-Okabe, Barak, & Su, 2024; Ren et al., 2024). Primarily, though, this thesis proposes that the black-box nature of these models is a good reason to keep working on alternatives.

A black-box model is a model where it is not quite clear how it operates, and how it transforms a certain input into the corresponding output. Considering that the most advanced language classification systems hold billions of parameters, this seems a given, and is accordingly supported by literature. Training is generally an automated process, while evaluation of these models usually happens through empirical testing, and not by taking a look at the model parameters themselves. This leads to three core problems:

- Interpretability: A black-box model teaches us nothing about the system it represents. By contrast, a more classical model can offer insights into the problem it is modeled on. Think, for instance, of a standard regression model, which assigns specific weights to its individual features. Or of a decision tree, which can be used to compute feature importance metrics for each given variable. In short, if the model is not transparent, we cannot comprehend the cause-effect relationship between input and output (Hassija et al., 2023). The need for interpretable models is further demonstrated by the development of black-box interpreters such as LIME (Ribeiro, Singh, & Guestrin, 2016) and DeepLIFT (Shrikumar, Greenside, & Kundaje, 2019).

- Explainability: Similarly, these types of models are hard to debug (Treude & Hata, 2023; Mozafari, Farahbakhsh, & Crespi, 2020; Da, Bossa, Berenguer, & Sahli, 2024). If a prediction fails, it is hard to understand why the prediction failed, if the failure is systemic, or what part of the model needs to be tweaked to prevent future failure. The state-of-the-art solution to these types of problems is usually to train the model with even more data and hope that the problem disappears after further evaluation.

- Reliability: since we cannot interpret or explain how a black-box model works, we can also not rely on it. Incorrect responses are particularly well-documented in LLMs, where they are also known as "hallucinations". While there is extensive study into the subject, it is not currently known how to prevent them (Huang et al., 2023). Even without hallucinations, one may question if a model trained on data scraped from the internet, encoding the knowledge of a general public, is suitable to conduct tasks that require an expert's touch. In addition, due to their often overly complex and obtuse network of parameters, black-box models are a ripe breeding ground for adversarial ML attacks, wherein malicious inputs can be used to force specific outcomes into existence or otherwise corrupt the model. (Wu et al., 2024; Qin, Li, Wang, & Wang, 2024)

As such, there is justified and continued interest into human-readable, rule-based classification systems. This is further undermined by multiple recent studies which attempt to establish such systems, for instance HEIDL (Sen, Li, Kandogan, Yang, & Lasecki, 2019) and GrASP (Lertvittayakumjorn, Choshen, Shnarch, & Toni, 2022).

## 2.3 POTATO

Another such system is TU Wien's POTATO (exPlainable infOrmation exTrAcTion framewOrk) (Kovács, Gémes, Iklódi, & Recski, 2022). It is a text classification framework that relies on defining and matching language patterns. POTATO relies on two core principles.

**Human-in-the-loop (HIL):** The system is making use of a human agent to establish its model. This has the advantage of leveraging human understanding to create rules. Particularly for problems that depend heavily on language understanding and domain

knowledge, this can be a great benefit. A rule might, from a statistical standpoint, generate a good performance score, but be otherwise meaningless. A human agent could identify these data artifacts and instead investigate and propose a more reasonable solution. A particularly knowledgeable human subject might be able to propose a sensible classification ruleset from scratch, although the framework is mostly designed around the idea of building a model through many iterations.

**Explainable AI (XAI):** POTATO's classification decisions are based entirely on pattern matching. Any decision that the classification model makes can be traced back to one or more specific rules. For false positives we can take a look at any of the triggering rules and try to make them more specific to exclude false matches. For false negatives we can take a look at the unmatched patterns and evaluate existing rules to determine if we need to adapt them or introduce a new one. The classification model itself, given a suitable performance, can also tell us a lot about the task at hand, as it will provide a set of meaningful rules that corresponds to the classification labels. It is also possible to evaluate each rule individually to see how they contribute to the system.

The framework comes with a graphical interface that allows a user to intuitively explore individual rules and experiment with their performances. For the scope of this thesis we will only interact with the library itself.

### 2.3.1 Syntactic and semantic graphs

Before we continue, it is important to understand syntactic and semantic graphs. In the same way that the previously-mentioned transformer models strongly rely on vector encodings, there are other ways to represent natural language, and these graph models are one such way. Traditionally, they will represent a given piece of text as a collection of nodes and edges, following some conversion system. POTATO implements three different conversion algorithms from the TUW-NLP (Recski, Lellmann, Kovacs, & Hanbury, 2021) library.

**Universal Dependencies** (UD) (de Marneffe et al., 2014) represent the grammatical structure of a given piece of language. In theory this system is built to be compatible with all known languages, as similar grammatical concepts will use the same tag across languages. In practice, the conversion is generated through a Stanza (Qi, Zhang, Zhang, Bolton, & Manning, 2020) pipeline. This pipeline provides a model that is trained on UD treebanks to provide tokenization, POS-tagging, lemmatization and a UD dependency parser. An example conversion of the sentence "Aspirin eliminates headaches" can be seen in figure 2.2

**Fourlang** (FL) (Kornai et al., 2015) is based around a dictionary of primitives, basic terms that can be used to define other, more advanced concepts in language. The aim is to provide a reductive conversion method that can express modern language through simple concepts. Likewise, relations are broken down to only three simple types: attribution & unary predication (0), subject (1) and object (2). In practice the implementation first generates a UD graph as described above, then reduces it to an FL graph using

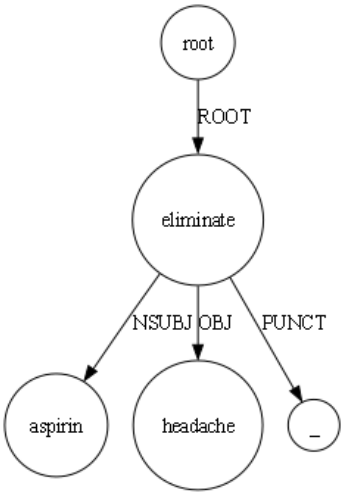



Figure 2.2: UD-conversion of "Aspirin eliminates headaches."

Interpreted Regular Tree Grammars (Ács Evelin, Ákos, & Gábor, 2019). An example conversion of the sentence "Aspirin eliminates headaches" can be seen in figure 2.3

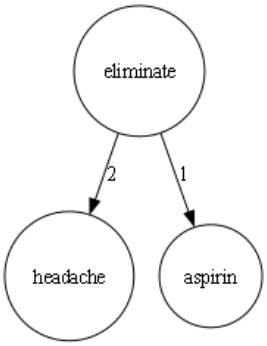



Figure 2.3: FL-conversion of "Aspirin eliminates headaches."

**Abstract Meaning Representation** (AMR) (Banarescu et al., 2013) models the semantic relationships between elements of a given piece of language. It is able to use PropBank frames to distinguish between homonyms of different meanings and comes with a set of predefined relations, from generic argument relations (:arg0, arg1, ...) to specific semantic concepts (:accompanier, :direction, ...). In theory, two grammatically different expressions of the same concept should turn out the same AMR graph. The implementation used by POTATO generates these graphs via the amrlib[1] package, which in turn produces its graphs through one of a selection of specifically trained transformer models. An example conversion of the sentence "Aspirin eliminates headaches" can be seen in figure 2.4.

[1]https://github.com/bjascob/amrlib

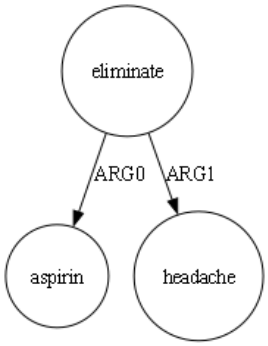



Figure 2.4: AMR-conversion of "Aspirin eliminates headaches."

**Penman notation** (Matthiessen & Bateman, 1991) is a way to represent all these graph formats in human-readable text form. It is a standard of notation that allows us to model any number of entities and their relations through a simple syntax. It is also possible to write penman-formatted graphs from scratch and convert them into graphs. POTATO uses penman notation to encode its matching rules. When evaluating the ruleset, these penman-formatted patterns are transformed back into graphs and passed on to the matcher, which then runs the comparisons. The below notation shows the UD representation of "Aspirin eliminates headaches" from figure 2.2 in penman format.

```
(u\_2 / eliminate
    :NSUBJ (u\_1 / aspirin)
    :OBJ (u\_3 / headache)
    :PUNCT (u\_4 / .)
    :ROOT-of (u\_0 / root)
)
```

### 2.3.2 Datasets

This thesis will focus on the CrowdTruth Cause dataset.

There are a couple of datasets that come included with POTATO. What this means is that POTATO offers precomputed graphs for these datasets, or there are documented experiments, or both. This gives a convenient point of comparison for any changes done to the system during this thesis, but it also eliminates a potential source of error, since the datasets are proven compatible. The following relation extraction datasets are shipped with POTATO:

**CrowdTruth** (Dumitrache, Aroyo, & Welty, 2018) is a medical classification dataset that comes split into two sub-sets: one for predicting a cause-effect relation between two entities, and one for predicting a treatment relation. The dataset has been labeled via crowdsourcing, with a score between 1 (CAUSE/TREAT) and 0 (NOT) reflecting the level of agreement between annotators. It comes pre-split into train/validation/test sets.

The entities are declared by indicating character begin and end positions for each entity. For an example of the data see table 2.1.

| # | Sentence | Score |
|---|---|---|
| 1 | The MCCUNE-ALBRIGHT SYNDROME is characterized by cafe-au-lait spots, PRECOCIOUS PUBERTY and fibrous dysplasia | 1 |
| 2 | Chan AS, Tang KC, Fung KK et al. SINGLE DOSE OFLOXACIN in treatment of UNCOMPLICATED GONORRHOEA | 0 |
| 3 | The abdomen may be tender to the point that an ACUTE ABDOMEN may be suspected, such as acute pancreatitis, appendicitis or GASTROINTESTIL PERFORATION | 0,7548294124 |
| 4 | The alterations found in bone matrix CONSTITUENTS IN OSTEOPOROTIC BONE RELATIVE TO CONTROLS suggest that in OSTEOPOROSIS and fractures, not only bone mass changes, but also bone quality changes play a role in bone strength. | 0,5303300859 |
| 5 | CLONUS appearing after ingesting potent SEROTONERGIC DRUGS strongly predicts imminent serotonin toxicity . | 0,8804509063 |

| # | term1 | b1 | e1 | term2 | b2 | e2 |
|---|---|---|---|---|---|---|
| 1 | PRECOCIOUS PUBERTY | 69 | 87 | MCCUNE-ALBRIGHT SYNDROME | 4 | 28 |
| 2 | UNCOMPLICATED GONORRHOEA | 71 | 95 | SINGLE DOSE OFLOXACIN | 33 | 54 |
| 3 | GASTROINTESTIL PERFORATION | 123 | 151 | ACUTE ABDOMEN | 47 | 60 |
| 4 | CONSTITUENTS IN OSTEOPOROTIC BONE RELATIVE TO CONTROLS | 37 | 90 | OSTEOPOROSIS | 108 | 120 |
| 5 | CLONUS | 0 | 6 | SEROTONERGIC DRUGS | 40 | 58 |

Table 2.1: Examples from the CrowdTruth Cause train set

**SemEval-2010 Task 8** (Hendrickx et al., 2010) is a semantic classification dataset. The task is to match a given pair of entities to one of ten possible relations. The entities are tagged directly in the input sentence. Each observation can only be assigned to one relation, a task for which annotators were given a set of guidelines. For an example of the data see table 2.2.

| # | Sentence | Label | Comment |
|---|---|---|---|
| 1 | "The system as described above has its greatest application in an arrayed <e1>configuration</e1> of antenna <e2>elements</e2>." | Component-Whole(e2,e1) | Not a collection: there is structure here, organisation. |
| 2 | "The <e1>child</e1> was carefully wrapped and bound into the <e2>cradle</e2> by means of a cord." | Other | |
| 3 | "The <e1>author</e1> of a keygen uses a <e2>disassembler</e2> to look at the raw assembly code." | Instrument-Agency(e2,e1) | |
| 4 | "A misty <e1>ridge</e1> uprises from the <e2>surge</e2>." | Other | |
| 5 | "The <e1>student</e1> <e2>association</e2> is the voice of the undergraduate student population of the State University of New York at Buffalo." | Member-Collection(e1,e2) | |

Table 2.2: Examples from the SemEval-2010 Task 8 train set

For an overview of the entity relation datasets refer to table 2.3. Table 2.4 provides a list of possible labels.

| Dataset | Topic | Total entries | T/V/T split | # classes |
|---|---|---|---|---|
| CrowdTruth Cause | Medical | 3990 | 80/10/10 | 2 |
| CrowdTruth Treat | Medical | 3983 | 80/10/10 | 2 |
| SemEval-2010 Task 8 | Generic | 10717 | 55/20/25 | 10 |

Table 2.3: Relation extraction datasets in POTATO

### 2.3.3 Building rulesets

The core unit of a POTATO classification model is the rule. A single rule is defined as a list of positive patterns, a list of negative patterns, and the associated label. The patterns are submitted in penman format. For example, to predict a cause relationship we might look for the positive pattern

```
(u_1 / cause
    :NSUBJ (u_2 / X)
    :OBJ (u_3 / Y)
)
```

| Label | # | Example |
|---|---|---|
| **CrowdTruth Cause** | | |
| CAUSE | 513 | Included are infections caused by the pathogenic organism **RICKETTSIA RICKETTSII** , which causes **ROCKY MOUNTAIN SPOTTED FEVER** |
| NOT | 492 | Bilateral vagotomy also inhibited both effects whereas atropine only reduced the **BRADYCARDIA** but the combition of **ATROPINE** and tertatolol suppressed the bradycardia. |
| **CrowdTruth Treat** | | |
| TREAT | 1225 | For this purpose 30 patients with **DUODEL ULCERS** were treated either with **RANITIDINE** alone (15) or together with bacampicillin (15), which was shown to be highly active in studies with ampicillin in vitro. |
| NOT | 644 | Ambulatory monitoring of blood pressure might allow for differentiation of patients with **TRANSIENT ELEVATED BLOOD PRESSURE** from those with more sustained **HYPERTENSION** |
| **SemEval-2010 Task 8** | | |
| Cause-Effect | 2094 | Rains and melting **snow** lead to Genesee County's biggest sewage **spill** of the year. |
| Component-Whole | 3579 | All **kangaroos** have a chambered **stomach** similar to cattle and sheep. |
| Entity-Destination | 253 | All **bookmarks** have been exported to a single **file**. |
| Product-Producer | 5048 | Ford's Dagenham **workers** still produced more **cars** for less pay than any other plant in Europe in the 1960s. |
| Entity-Origin | 865 | The popular **definition** is rooted in an editorial **error**. |
| Member-Collection | 98 | In the corner there are several gate captains and a **legion** of Wu **crossbowmen**. |
| Message-Topic | 1046 | The **play** reflects, among other things, **questions** about the nature of political power and the dilemmas facing royal families |
| Content-Container | 3839 | The **stomach** contained a small amount of bile-stained **acid fluid**. |
| Instrument-Agency | 6115 | A Management Unit **personnel** marked private lines with blue **paint**. |
| Other | 3334 | The **farm** is participant in **forestry** and there have already been planted around eight thousand plants. |

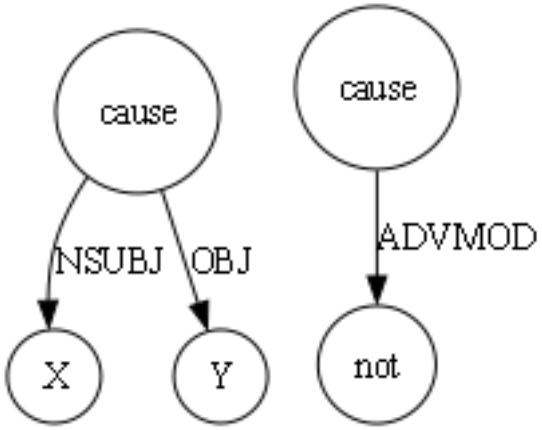
Table 2.4: Classes in POTATO's relation extraction datasets

To prevent false matches, such as "X does not cause Y", we could then add the negative pattern

```
(u_1 / cause
    :ADVMOD (u_4/ not)
)
```

The full rule (illustrated in figure 2.5) would be submitted as

```
[
    # positive rules
    ["(u_1 / cause :NSUBJ (u_2 / X) :OBJ (u_3 / Y))"],
    # negative rules
    ["(u_1 / cause :ADVMOD (u_4/ not))"],
    # label
    "CAUSE"]
)
```




(a) "X causes Y" (b) no cause

Figure 2.5: UD representations of example rules

Now POTATO will iterate through all graphs in the dataset. For any given graph, if all positive subgraph patterns can be found, and if none of the negative patterns can be found, POTATO predicts the "CAUSE" label for the given sample.

We can add as many such rules as we like, and only one of them needs to evaluate as true for the assignment to trigger. This provides us with a couple logical combinations. A single rule gives us the ability to chain multiple subgraph patterns via AND condition, meaning all of the matches need to be found. Similarly, the negative patterns grant us the power of negation (none of them may match). If we add another rule, it is effectively adding an OR condition - if any rule is true, the corresponding label is predicted.

In addition, POTATO implements the TUW-NLP library feature to define paths with indistinct numbers of inbetween nodes. This is done by inserting one of three functions into the penman notation:

- path(X,Y) matches with a directed path of any length from node X to node Y
- undirected(X,Y) matches with a path of any length from node X to Y regardless of direction
- 3(X,Y) matches with a directed path from X to Y where both are at most three steps away from each other, meaning no more than three edges are traversed to finish the path. Note that this notation can be used with any number, so 5(X,Y) will do the same with 5 steps, etc.

For instance, if we wanted to define a rule that assigns the label "CAUSE" to any graph that contains a path between nodes "cause" and "X" and a path between nodes "cause" and "Y", we could define it as such:

```
[
    # positive rules
    ["path((u_1 / cause), (u_2 / X))",
    "path((u_1 / cause), (u_3 / Y))"],
    # negative rules
    [],
    # label
    "CAUSE"]
)
```

Note that the two "cause" nodes are not guaranteed to be the same. This is elaborated on in later chapters.

Finally, POTATO can interpret RegEx patterns. For instance, if we don't care about the label of a given node or edge, we could just write a penman pattern that contains the label ".*" in its place, meaning it will match with any node or edge, respectively. For example, let's say we have this rule:

```
[
    # positive rules
    ["(u_1 / .* :.* (u_2 / X) :.* (u_3 / Y))"],
    # negative rules
    [],
    # label
    "CAUSE"]
)
```

This pattern will match with any graph where nodes "X" and "Y" share the same parent. Similarly, if we wanted to be a bit more specific, we could define this rule:

```
[
    # positive rules
    ["(u_1 / cause|activate|result :.* (u_2 / X) :.* (u_3 / Y))"],
    # negative rules
    [],
    # label
    "CAUSE"]
)
```

which would match with any parent node that has either the "cause", "activate" or "result" label.

Interestingly, POTATO has a feature that does this last step automatically, called refinement. If we assign a RegEx wildcard ".*" to a pattern, we can ask POTATO to refine this label for us. POTATO will then conduct a pattern search through the dataset and evaluate each matched subgraph pattern individually, using the matched label instead of the wildcard. If this evaluation yields at least 90% accuracy and non-zero recall, it will add the term from that subgraph pattern to a disjunctive regular expression. This regular expression will then finally replace the wildcard in the refined node or edge label.

### 2.3.4 Entity Tagging

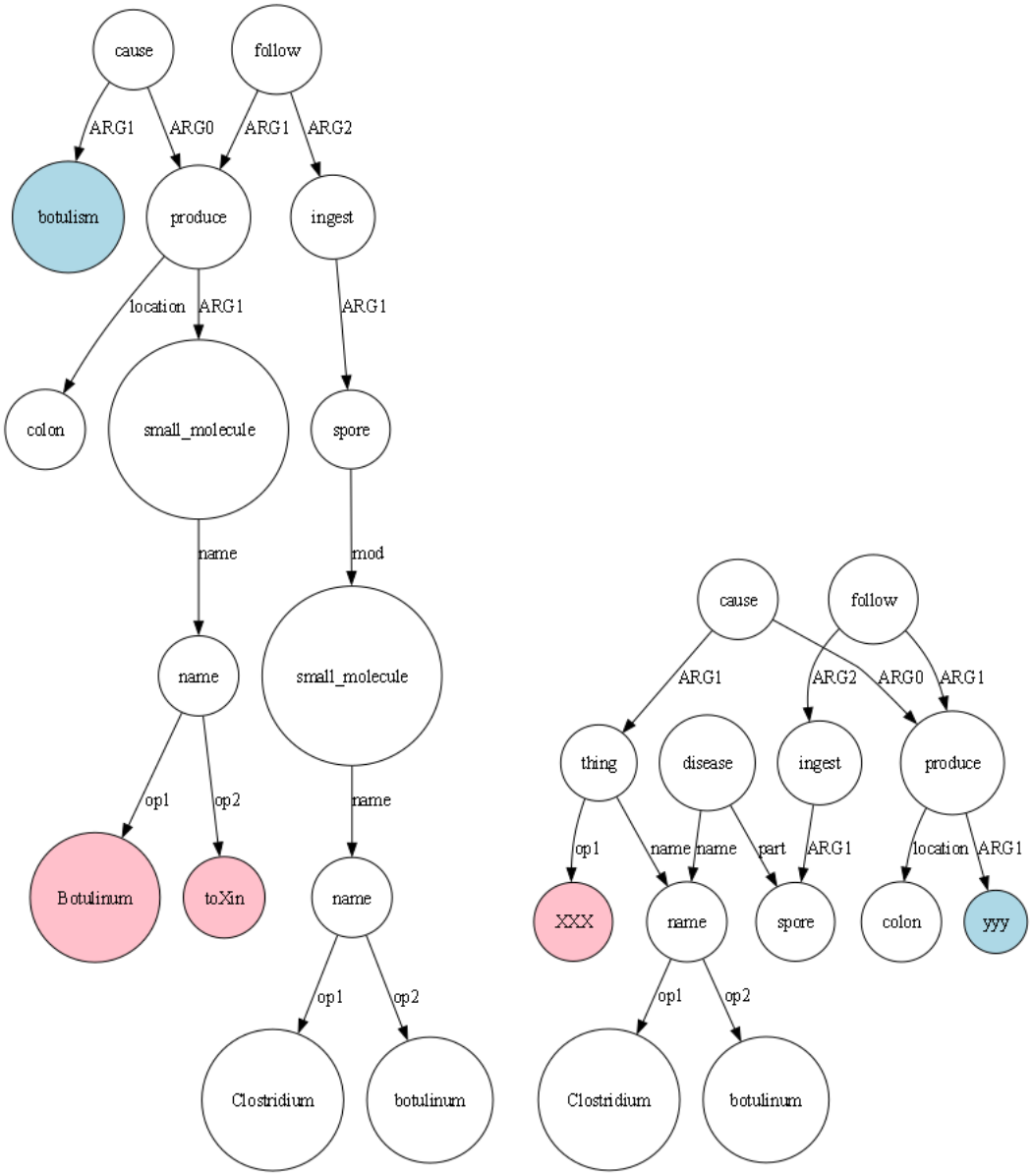
Obviously, if we want to work on entity relation problems, we need some way to mark entities in the generated graph representations. This is one issue with POTATO. The current method is designed like this:

1. Load the source sentence
2. Replace all occurrences of entity 1 with XXX
3. Replace all occurrences of entity 2 with YYY
4. Proceed normally to graph conversion with tokenization, POS-tagging, etc

This is a somewhat destructive approach to marking entities, and in theory leads to two problems:

As covered earlier, graphs are constructed using models that encode a certain understanding of the human language. These models may have been built and validated using painstakingly annotated treebanks, trained on copious amounts of reference text, or both. It stands to reason that by removing words or even entire phrases from the input text and replacing them with a single placeholder token, we rob these entities of important context, which can create a syntactic or semantic graph that is not representative of the original text. Figure 2.6 illustrates this. We can see clearly that conversion of the given sentence results in a vastly different type of structure depending on whether the

tokens were replaced or not. Since the goal of POTATO is to use human understanding of language to build rulesets, these unintended changes to graph structure may depress performance of any attempted classification system.



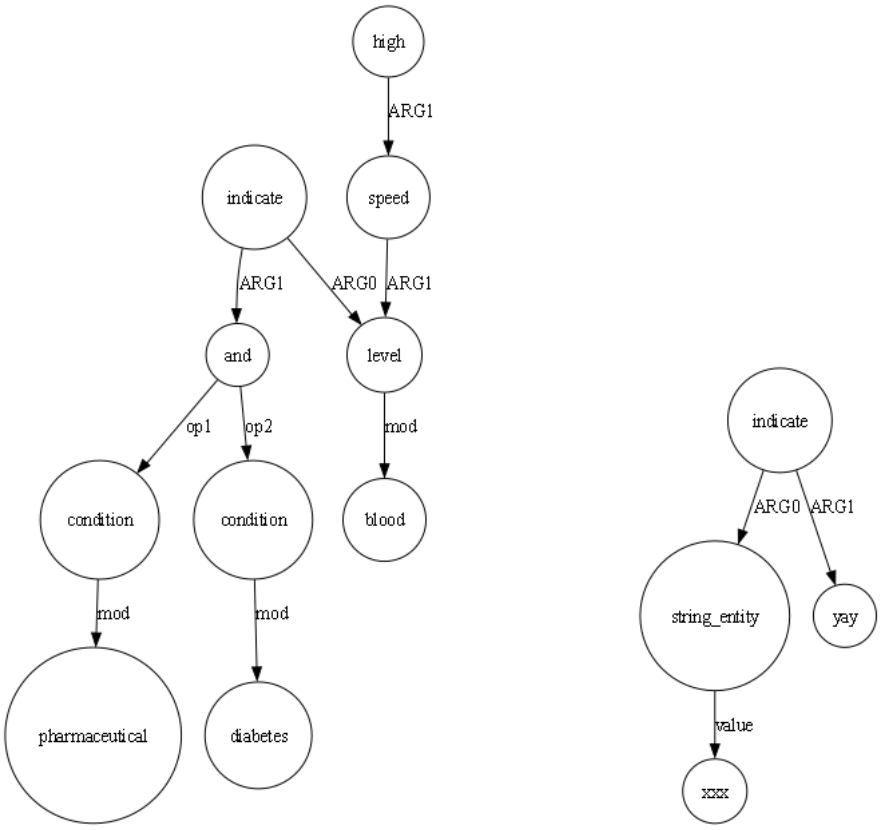

(a) Original (b) Entities replaced

Figure 2.6: AMR representations of "BOTULISM caused by production of BOTULINUM TOXIN in the colon following ingestion of spores of Clostridium botulinum." Entities highlighted for comparison.

Even more glaringly, we can observe that some graph converters omit the entity tag entirely. Take for instance CrowdTruth Cause sentence

#1300 "A HIGH FASTING BLOOD SUGAR LEVEL is an indication of PREDIABETIC AND DIABETIC CONDITIONS."

We can observe in figure 2.7 that, after conversion, the entity "YYY" is absent from the final tagged graph (as the transformer likely converted "YYY" to "yay"). This, of course, puts an entire representation system into question.

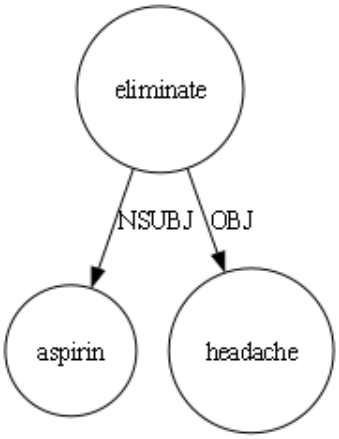



(a) Original (b) Entities replaced

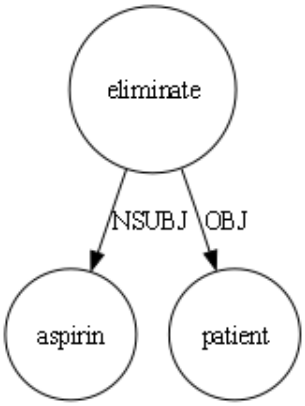
Figure 2.7: AMR representations of "A HIGH FASTING BLOOD SUGAR LEVEL is an indication of PREDIABETIC AND DIABETIC CONDITIONS."

Additionally, it inhibits our potential to create meaningful patterns. Not only could the entity nodes contain vital information for a classification decision, the fact that certain terms are part of the entity in itself could be of interest to a user. For example, consider figure 2.8, a well-known example for polysemy in relation inference as seen in Levy and Dagan (2016). Clearly the two given input graphs carry very different meanings. However, the current conversion algorithm translates them into the same output. Therefore it is impossible to distinguish between them at the point of entity relation classification.

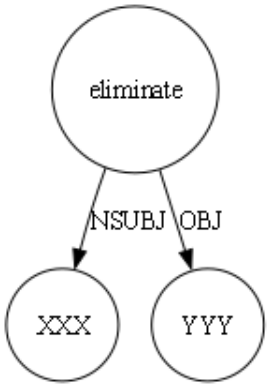
It bears repeating that a key component of working with POTATO is that the process makes use of human understanding. While a domain expert may not always be at hand, some level of expertise can be approximated by using knowledge graph stores or similar semantic repositories that model what we know about a certain scientific field. For instance, take a medical database that stores known diseases and their symptoms. Neither an expert nor a properly aligned knowledge graph could use the entity-tagged representation in figure 2.8 to distinguish between the case where, clearly, a drug that eliminates headaches is a treatment, and a drug that eliminates patients is a health hazard.




(a) "Aspirin eliminates headaches"




(b) "Aspirin eliminates patients"




(c) "XXX eliminates YYY"

Figure 2.8: Two different graphs converge after entity tagging

## 2.4 Aim of this work

As we can see, POTATO is still a relatively young framework, under ongoing development, and there are likely some opportunities of improvement. We have also made the case that there is a justified interest in developing such a HIL XAI language classification system.

Therefore, this thesis is an opportunity to investigate, identify and address technical limitations of the POTATO framework. This will be done through a practice-oriented approach where we use POTATO to build an example ruleset for a classification task. We follow the intended process of iteratively building a new set of predictive subgraph patterns, with the aim to identify medical cause-effect relationships in the CrowdTruth Cause dataset. Our goals for this first step of the thesis can be summed up through the following research questions:

RQ 1 What are the shortcomings that we can identify in POTATO's current pattern matching capabilities that prevent us from establishing rulesets that reach better precision and recall scores on a classification problem, as demonstrated by the CrowdTruth Cause dataset?

RQ 2 What changes would we need to introduce to POTATO so that we could address these problems and enhance our precision and recall scores?

In addition, we will put one of our proposals to the test. In the course of this thesis, we will implement a new system for marking entity relation participants. We theorize that a non-destructive method of tagging these entities will lead to a more accurate graph conversion process and allow us to leverage domain knowledge by writing rules that take entity node labels into consideration. The goal of this additional experiment is summed up in

RQ 3 What is the precision and recall score that we can reach on the CrowdTruth Cause dataset when using the new entity tagging system?

CHAPTER 3

# Methods & Proposed Changes

## 3.1 Methodology

To satisfy RQ 1-2 we want to build a ruleset as intended by the POTATO framework: manually define subgraph patterns and iteratively work through sets of rules to establish a classification model for medical entity relations.

At its core, this is still a classification task as is standard in data science. This thesis therefore adheres closely to common norms and principles of data science projects. In the following section, we illustrate how our process of building the example ruleset can be mapped to an established scientific method such as CRISP-DM (Wirth & Hipp, 2000):

1. **Business Understanding**
   It is important that we first define the actual goal of the process. One key use of POTATO is for linguistic research, by building explainable classifier systems on language tasks. The aim is not to be a top-of-the-line benchmark-setter. As such, we determine that our main concern is a transparent and reproducible process, by which we can illustrate advantages and technical limitations of the framework's current implementation. As POTATO is meant to be used in iteration, we also know we will need to define a stopping condition for the model building process.

   The creators of POTATO provide various reference implementations of experiments on language classification tasks, which is another key component to understanding the business case of the framework, as it clearly demonstrates the intended way of handling the system for a variety of NLP problems.

2. **Data understanding**
   We work with the CrowdTruth Cause dataset, as explained in section 2.3.2. On one hand, there are some base considerations with every dataset, and perhaps with

crowd-labeled datasets in particular: how reliable are the labels? How accurate is the entity marking? How meaningful are the text snippets? As an example take the sentence

#35 "280 , 281 , 283 , 285 Altetive for treatment of GLANDERS + caused by B MALLEI"

Samples that contain a significant portion of meaningless text and perhaps even outright typing errors and misspellings may be hard to parse for a rule-based language converter. Wrongly declared entity boundaries may make it impossible for the converting mechanism to correctly mark the entities. Inaccurate labels may falsify the performance of otherwise predictive or inappropriate rules. We know that these factors may play a role in our dataset, and it is important to keep that in mind when analyzing the performance of our model.

On the other hand, getting a feel for the data is an important foundation for establishing the classification model. POTATO is a human-in-the-loop system; classification decisions are, by design, based on a human understanding of language. As we document in section 4.1, an important first step of building the classification system is to draw samples from the dataset and try to classify them in plain English. This gives us a good idea of re-occurring themes and ideas, as well as the general reliability of the dataset. It sets the stage for the modeling process, where we try to match our naive and idealized classification ideas with the reality of POTATO's technical capabilities.

3. **Data preparation**
The dataset comes pre-split into train/validation/test sets. To avoid any data leakage, we strongly adhere to these splits by the following logic:

- The **train set** is used in the iterative process to come up with subgraph patterns for the classification model, and to check the performance of individual rules and patterns as we add them to the ruleset.
- The **validation set** is used to occasionally test the generalization potential of a work-in-progress ruleset, before going for another iteration of rule generation on the train set.
- The **test set** score is only calculated once, at the end of the thesis, to provide a performance metric with minimal interference in the process.

A key component of data preparation is the conversion from text to graph. Much of this process is covered implicitly by POTATO, which takes care of the required preprocessing methods (such as tokenization, POS-tagging, lemmatization, parsing) for each graph system, as explained in section 2.3.1. To provide maximum flexibility when building the model, we parsed the entire dataset into all three of POTATO's supported graph systems: UD, FL and AMR.

One step that POTATO doesn't take care of is the marking of entities in entity relation problems. The current reference implementation provided by POTATO replaces entity labels with placeholder text, as shown in 2.3.4, and we follow this standard.

There is a significant level of interplay with step 2. As we see, a core part of data preparation is to convert source text into graph representations. As we generate these graphs, we also need to examine them like we did with the original source texts, to understand re-occurring structures and common patterns for the modeling step.

4. **Modeling**
   POTATO is a human-in-the-loop system, and as such, the framework is meant to be used to build classifiers in an iterative process. We will attempt to predict a medical cause relationship through a set of rules (for a summary of POTATO's rule system, see section 2.3.3). With each iteration of the build process, we will analyze the ruleset's train performance, see how it generalizes into the validation set, analyze false positives and negatives in the train set, and use any gained insights to improve the model step-by-step.

   It should be noted that we can and will analyze rules individually. The fact that we can draw a clear line from each individual model feature (as represented by the patterns) to its performance contribution is a big advantage of POTATO over many other classifiers, which only return a summary performance for the entire model and require additional methods to compute individual feature importance metrics. We also do not have to bother with any hyperparameter optimization.

   Eventually this ruleset should converge. When we can no longer find any meaningful patterns to add to the model, it is considered final. As one might imagine, due to the iterative nature of the process, there will be heavy interaction between steps 4 and 5.

5. **Evaluation**
   After an initial set of rules has been created, we compute train and evaluation metrics, particularly for precision and recall. A low precision score will let us know that some of our rules are too imprecise and need to be defined more accurately. A low recall score will let us know that our ruleset does not cover enough ground. This could be because our rules are overly specific, or it could be that the set of graph representations simply does not provide many common patterns that POTATO can currently make use of.

   To get a better picture of these concerns, each round of evaluation includes an investigation of false positives and false negatives, to address concerns in regards to precision and recall, respectively. This also allows us to look for reoccurring themes in classification errors and determine if these are a flaw of the classification model or the framework itself.

6. **Deployment**
   At the end of the process, we evaluate the classifier on the test set and publish the scores in chapter 5. Appendix II provides a documentation of the complete ruleset, along with visualized match patterns (where applicable), penman notations, example sentence matches and individual train and validation performance scores for each rule.

To analyze the performance of our entity tagging system as outlined in RQ3, we will repeat the process, with two important distinctions:

1. We will rerun the previously built system, but with the dataset and all rule patterns converted to the new entity tagging system
2. We will introduce additional rule patterns that could not previously have been written

Here, we vaguely lean on concepts formalized by the PRIMAD (Braganholo et al., 2016) model. At first, we are essentially reproducing the original experiment. Our source data doesn't change, our classification approach doesn't change, our tech stack doesn't change and our ruleset doesn't change. The classifier is completely deterministic and will always deliver the same result on the same data. The only aspect that changes, the primed component so to speak, is how entities are encoded into the language graphs, which we theorize will lead to a more accurate representation of the language contained in the dataset. It stands to reason that any improvements to the classifier's performance are due to this change to the entity tagging system.

In a second step we then reproduce the experiment again, following the same line of argumentation, but this time we add a set of rules that could previously not have been expressed. We do this by focusing our rule-making logic on the entity labels themselves. Thereby we can demonstrate that the new system allows us to access information that was previously destroyed during the conversion process, and that we can use this information to write useful patterns.

In summary, our focus when working on RQ3 is to keep the amount of moving parts to a minimum, and re-evaluate the system as a whole after each major step. This should allow us to isolate the effect that these changes have on our classifier performance, and thereby provide a reasonable assessment of how useful the new entity tagging system is for our relation extraction task.

## 3.2 Proposed changes to POTATO

As part of this thesis we have investigated the capabilities of the POTATO framework on the CrowdTruth Cause dataset (see section 4.1). We have come to a few conclusions where the system might be lacking in features and proposals to enhance them.

### 3.2.1 Consistent node IDs

The first opportunity for improvement is to introduce a consistent memory of node IDs within a chain of positive rules. Imagine we are building a system to classify medical causes between entities and come up with the following rule:

```
path((u_1 / cause), (u_2 / XXX)),
path((u_1 / cause), (u_3 / YYY))
```

One would expect that this matches any graphs wherein a node with the label "cause" connects to both entities. However, it is important to note that the cause node in pattern 1, despite having the same ID, does **not** have to be the same node as in pattern 2. Imagine the following sentence:

"While XXX causes relief, patients feel that YYY causes pain."

If we examine its UD representation in figure 3.1, we can see that the rule would still trigger, although we may not have expected it to.

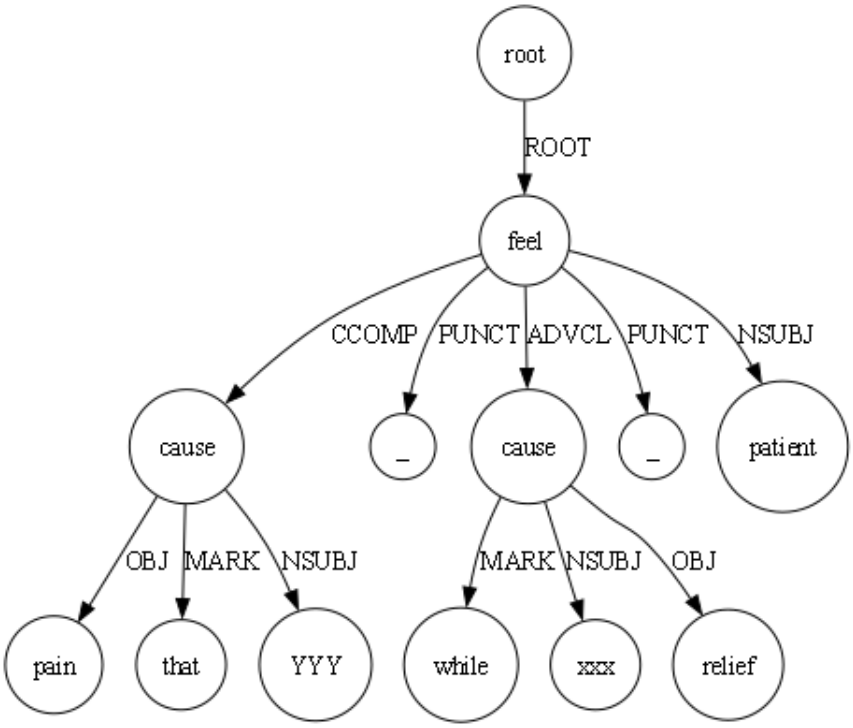



Figure 3.1: UD representation: "While XXX causes relief, patients feel that YYY causes pain."

### 3.2.2 Syntactic sugar

There is a list of potential improvements to the way that rules are expressed in POTATO which we would summarize as "Syntactic sugar". The reason is that a lot of these ideas are technically possible in the current system, if you use workarounds. However, not all of these workarounds may be accessible via the POTATO web interface, which arguably is the main access point for a less technically apt user.

First of all, the path system does not offer a few conveniences that might be useful:

- Designate a node that must or must not occur within the path
- Designate an edge that must or must not occur within the path
- Designate a subgraph that must or must not occur within the path

These items also extend to undirected or step-limited path functions.

A similar nice-to-have would be the option to create rules of equivalence for types of patterns. Assume we do not want to use a path function but instead we want to specify a pattern. Now take the graphs in figure 3.2. With the current system we would have to define one rule for each of these patterns, due to the varying numbers of distance between the "cause" node and the entities. However, the idea could be summed up in one rule: "Something to do with X causes something to do with Y". Assume for instance we could define

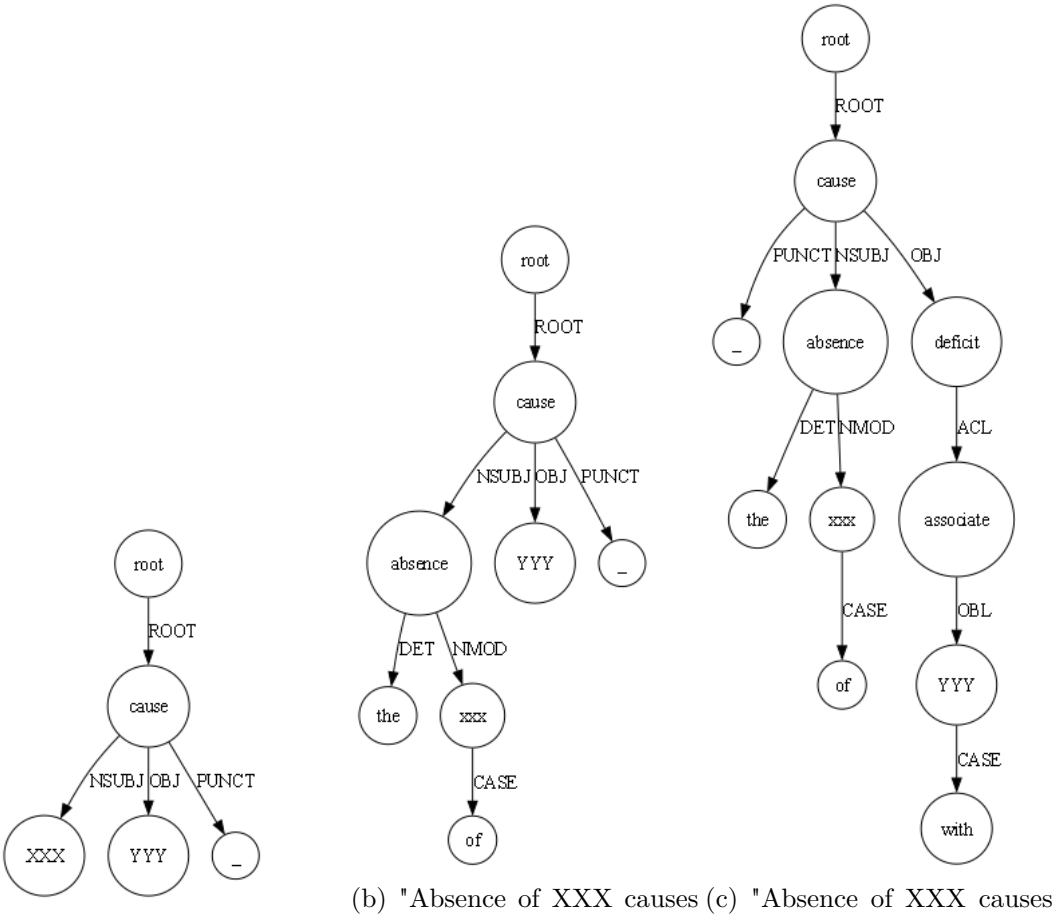
```
(u_2 / XXX)
```

and

```
(u_1 / absence)
    :NMOD (u_2 / XXX)
```

as identical patterns. If we could store these definitions, this would likely cut down on the need for creating granular rules for every single possible combination of modifiers and avoid an arbitrary number of rules where the core idea might be expressed in one. Perhaps these rules could even be defined by permitted connector nodes, along which equivalence is built recursively. As we have already seen, the path function is not a suitable substitute, since we cannot clearly define that both paths need to originate from the same node.

Next, assume we want a single rule that can find any structure of graph such as 3.3, meaning we don't care which branch contains which entity. The structure in these two graphs is exactly the same, except for the swapped entity labels. However, if we define a rule such as



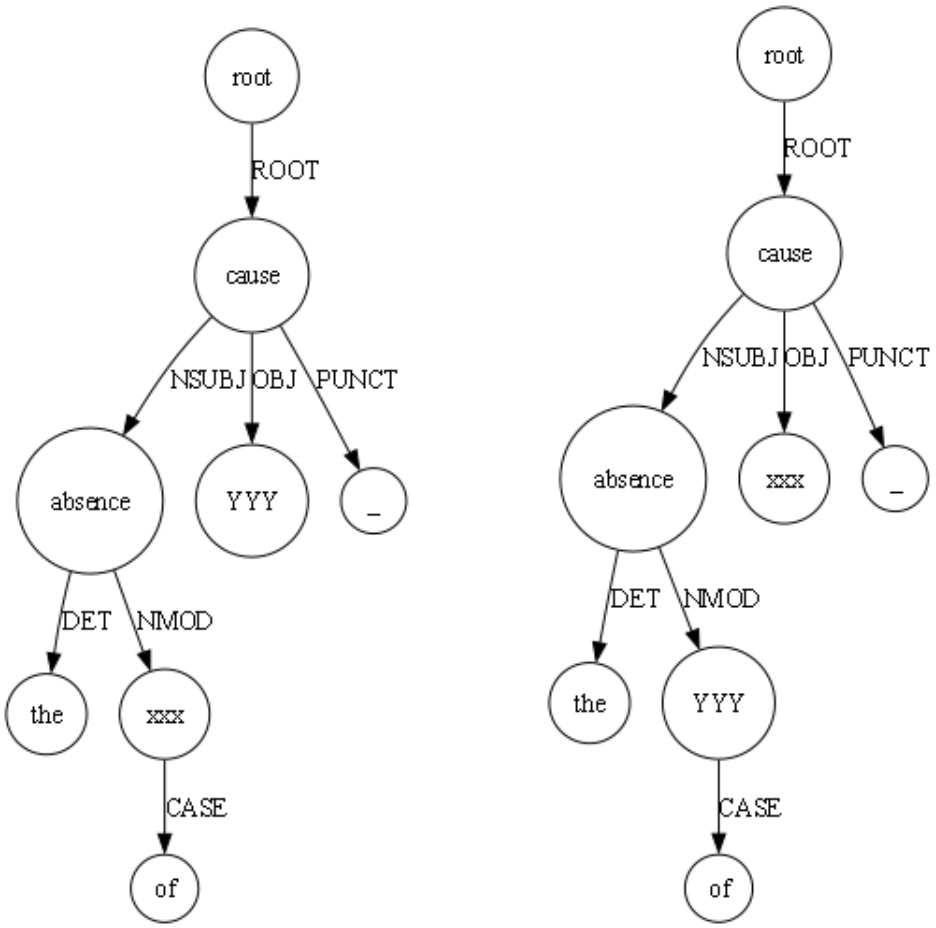

(a) "XXX causes YYY." (b) "Absence of XXX causes YYY." (c) "Absence of XXX causes deficits associated with YYY."

Figure 3.2: UD representations of similar concepts

```
(u_1 / cause) : (u_2 / XXX|YYY),
path((u_1 / cause), (u_3 / YYY|XXX))
```

then we would also find a graph like shown in 3.4, which would be a match we probably would not initially expect. This is because the same entity node "YYY" matches twice: it is a direct descendant of the cause node and therefore a path also exists between them. Introducing some sort of notation that lets us designate two entities as being different from one another without having to specify which is which might be useful for cases like these. But to be clear, the workaround here is particularly easy: we can simply write this kind of rule twice, but with the labels flipped.

Finally, while POTATO currently integrates three different language representation systems (UD, FL, AMR), it does not offer the possibility to write a single rule system that uses all these representations, as the pattern matcher generally expects one consistent set of graphs to compare against. As we will see in chapter 4, there are some unexpected




(a) "Absence of XXX causes YYY." (b) "Absence of YYY causes XXX."

Figure 3.3: UD representations of mirrored expressions

pitfalls when dealing with UD representations. From that experience, we would propose that a mixed-system ruleset has its benefits. Perhaps a user could pass various graph sets to the matcher, and establish a mapping for each rule, or even for each pattern within a rule.

### 3.2.3 Entity tagging system

In this thesis we propose the use of a new entity tagging system. Currently, POTATO replaces entities with placeholder labels before tokenization. As we have seen in section 2.3.4, this can lead to some severe problems when converting the data into graph form and when attempting to build a meaningful ruleset. Instead, it seems reasonable to convert the original text into its proper graph form, and then mark entity nodes with a special attribute.

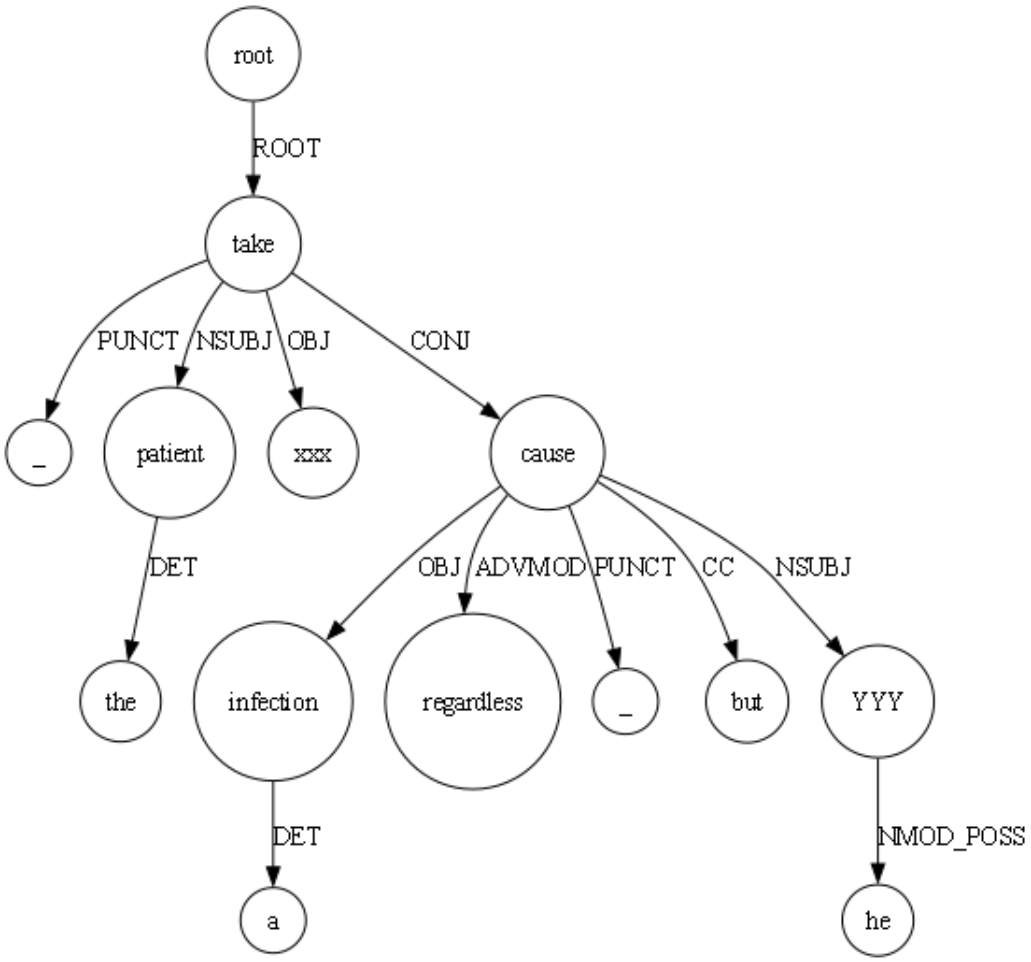



Figure 3.4: UD representations of: "The patient took XXX, but his YYY caused an infection regardless."

The technical implementation can work in the following ways for each graph system:

- UD: the UD conversion algorithm returns, amongst other things, a data structure that precisely maps graph nodes to character position in the source text. This mapping can be accessed to tag the correct nodes. Identifying which node is an entity is trivial, if the dataset provides this information correctly.
- FL: Fourlang graphs are built from UD graphs. Through this process, the UD node IDs are retained in the resulting FL nodes. The idea is to use this information to build a mapping that links FL nodes to UD nodes. Since we know which UD nodes are entities, we can then tag the respective FL nodes. It should be noted that FL graphs are usually a reduction of the UD representation. In theory, entities should be preserved during this translation. In practice, the conversion algorithm

will discard individual nodes, and some of these nodes may be entities. There is no guarantee that all UD entity nodes can be mapped to a node in the FL graph. As a result, some graphs may be missing one or both entities.

- AMR: these graphs are built through a transformer-based text generation model. The algorithm does not provide a precise mapping out of the box, but POTATO implements the amrlib[1] RBWAligner, a rule-based aligner that returns mappings from nodes to tokens. The mapping from character position to token has to be established separately. We achieve this by re-implementing the same spacy[2] tokenization step that amrlib is using and accessing the resulting token set. Through these two mappings it is then possible to link nodes to entities. However, it is possible that some entities go missing, particularly because the generative transformer is not completely reliable.

Each entity node will receive a new attribute, "entity", with a value of either 1 or 2 that corresponds to its assignment in the source dataset.

In a next step, the new graphs need to interface with the POTATO rule system, which relies on penman notation to pass the graph patterns on to its matcher. The current implementation of this process does not take attributes into account. Therefore, we also update the penman conversion method, by modifying the following two steps:

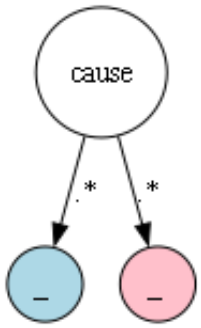
1. When converting from graph to penman, we resolve the entity attribute by generating a new relation :entity to a node with a label of 1 or 2, depending on its entity assignment.
2. when converting from penman to graph, this process is reversed: all entity relations are stored in their respective nodes as attributes, while the placeholder entity edges and nodes are deleted.

This process is illustrated in figure 3.5.

The pattern matcher is also updated to check for the entity attribute. Two nodes only count as equal if either none of them have an entity attribute, or both have the same value.

[1] https://github.com/bjascob/amrlib
[2] https://github.com/explosion/spaCy

```
{name: "cause", token_id: 1},
{name: ".*", token_id: 2,
 entity: 1},
{name: ".*", token_id: 3,
 entity: 2}
```




(a) Graph and node dictionary

```
(u_1 / cause
    :.* (u_2 / .*
        :entity (entity_0 / 1)
    )
    :.* (u_3 / .*
        :entity (entity_1 / 2)
    )
```

(b) Penman notation

Figure 3.5: Example of a graph to penman conversion using the new entity tagging system

CHAPTER 4

# Experiments

## 4.1 Building a manual ruleset

The main purpose of creating a manual ruleset was to see if, through the process of creating such a classification system, we could identify any potential for automation or refinements to the rule writing system. It was an important first step in our work on this thesis that allowed us to evaluate the current capabilities of the framework. The choice was to work with the CrowdTruth Cause dataset, through an iterative process defined as such:

1. Select a random list of indices from the CrowdTruth Cause train set.
2. For each index, look at the given sentence. If it is a positive label, try to classify, in plain english, what makes this an example of the label. If this cannot be identified, skip the sample.
3. Further formalize these classifications by highlighting key segments and summarizing them, where possible, into a regular expression. This is a useful exercise. POTATO itself can match node and edge labels using regular expressions. This process might provide a good idea where similar rules should be able to be grouped together.
4. Repeat steps 2-3 until convergence, i.e. there is a number of consecutive rules for which no new rules can be defined, or all new rules seem overly specific and meaningless.

An excerpt of the process can be seen in table 4.1, while the full result is attached in Appendix I under table I.1. Regular expressions are encoded in the node and edge labels as can be seen in appendices II and III.

| ID | Sentence | Key section | Plain speech rule |
|---|---|---|---|
| 657 | Abstinence from YYY and food eating processes causes XXX in those with eating disorders. | Abstinence from YYY [...] causes XXX | direct cause |
| 1634 | Carpal Tunnel Syndrome affects the hands since it is an YYY that results in motor and XXX nerve. | YYY that results in [...] XXX | direct cause |
| 3165 | Treatment of dermatophyte infections of the toeil or fingeril (onychomycosis, XXX caused by susceptible YYY | XXX caused by [...] YYY | direct cause |
| 1025 | Treatment of XXX in severe YYY does not always result in clinical improvement in the patient's central nervous system. | Treatment of X in [...] Y | implied symptom |
| 746 | XXX may result from abnormal epinephrine and norepinephrine YYY or from cortisol excess (ACTH secreting tumors. | XXX may result from [...] YYY | direct cause |

Table 4.1: An example of positives and rules from the manual ruleset creation process

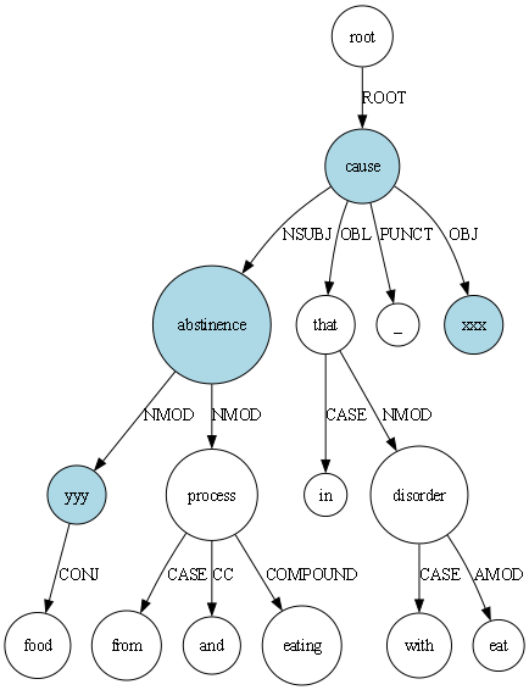
We then converted this set of rules into UD patterns. We expected that each major type of plain speech classification would represent one set of similar ideas of graph, perhaps with some different individual patterns for grammatical aberrations that ended up expressing the same idea. Our aim was to avoid single-hit rules if they seemed overly specific and not generalizable. We wanted a solid foundation more than a collection of data artifacts. After all, table 4.1 shows that many grammatically different constructs can be summarized by the same core idea. This should, in theory, be a strength of the combined pattern matching and RegEx wildcard approach of POTATO.

However, the process of creating these patterns has unearthed the need for many different types of patterns to express similar ideas, even for just tiny grammatical differences. This is due to the nature of the UD parser. For instance, take the sentences

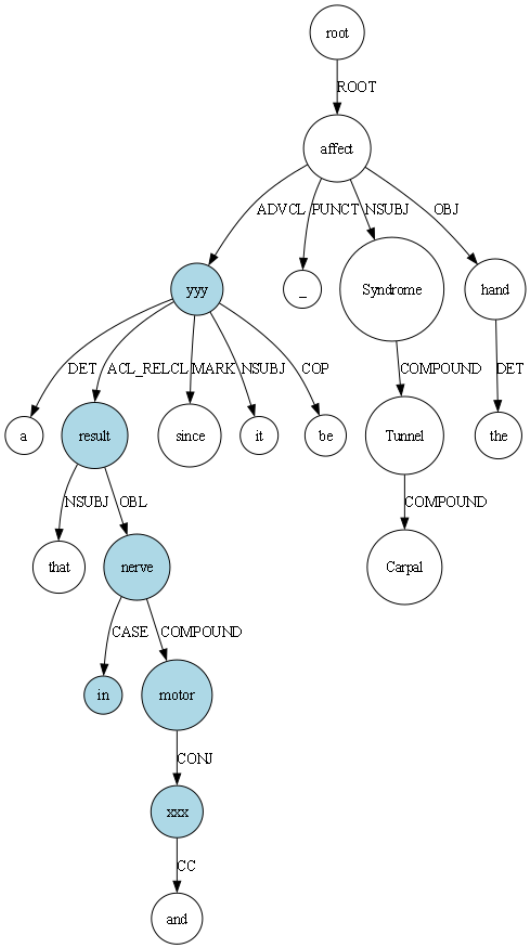
#657 "Abstinence from YYY and food eating processes causes XXX in those with eating disorders."
#1634 "Carpal Tunnel Syndrome affects the hands since it is an YYY that results in motor and XXX nerve."

and consider that we could express the key verb node through a regular expression $(cause|result)$, which makes the key segments of these sentences seem very similar

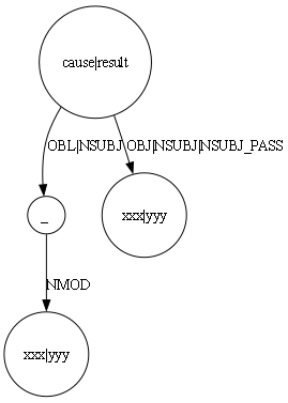
Abstinence from YYY [...] causes XXX
YYY that results in [...] XXX



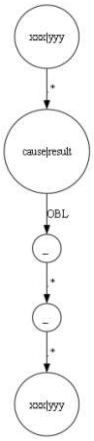

(a) "Abstinence from YYY and food eating processes causes XXX in those with eating disorders."




(b) "Carpal Tunnel Syndrome affects the hands since it is an YYY that results in motor and XXX nerve."




(c) This rule matches the pattern on the left




(d) This rule matches the pattern on the right

Figure 4.1: Comparing the key subgraph patterns of two sentences.

Yet, when we take a look at how these segments are expressed in the subgraphs of the respective UD representations, as shown in figure 4.1, we will note that the structural arrangements are noticeably different and therefore need a different pattern logic to be expressed, as reflected by figures 4.1c and 4.1d respectively

While building the ruleset, it also turned out that some rules had to be written in an extremely specific way to match, which left doubts about the generalization capability of the classifier. Take for instance sentence

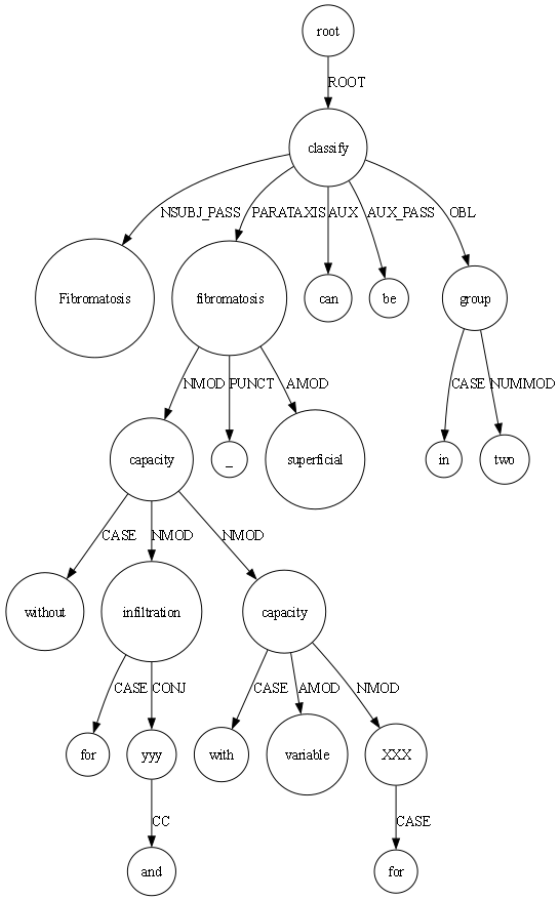
> #211 "Fibromatosis can be classified in two groups: Superficial fibromatosis without capacity for infiltration and YYY with variable capacity for XXX"

One would assume that the correct subgraph is the one for "YYY with variable capacity for XXX", or perhaps some pruned version of this, as shown in figure 4.2b. However, as we can see from the full version of the sentence (figure 4.2a), the correct subgraph pattern would actually be as shown in figure 4.2c. The context within which these sub-strings occur can heavily influence the structure of their related subgraph patterns.

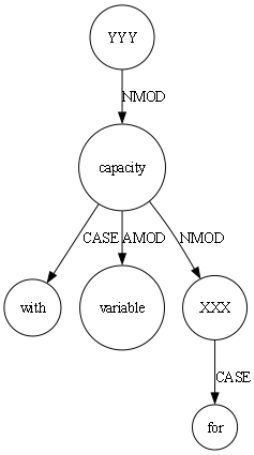
When building rules it is careful to avoid falling into the trap of writing a rule that seems to perform well in the classification metrics but is actually meaningless. Take for instance the rule

```
path(
    (u_13 / cause|result|occur|characterize|
            implicate|involve|reactivation|
            begin|suggest|exacerbate|begin|
            increase|precipitate|indicate|
            associate|experience),
    (u_11 / XXX|xxj|YYY|yyj)
)
```

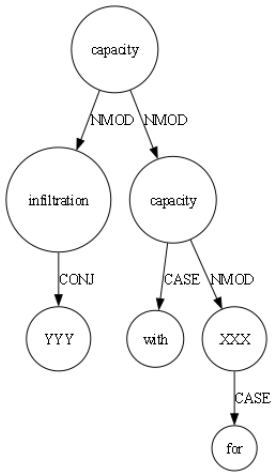
This will in effect look for any connection between a node of cause synonyms and any entity. Technically this doesn't give us too bad of a performance score (prec = 0.636935, recall = 0.42005, f1 = 0.506241), but really the rule "one cause node connects to an entity" is meaningless for classifying relationships between entities.




(a) "Fibromatosis can be classified in two groups: Superficial fibromatosis without capacity for infiltration and YYY with variable capacity for XXX"




(b) The expected subgraph pattern to match: "YYY with variable capacity for XXX"




(c) The actual subgraph pattern to match

Figure 4.2: Expected vs actual matching subgraph-pattern

After the initial creation of our manual ruleset, we further analyzed false positives and false negatives, using the information to restrict the scope of existing rules or expand the ruleset, respectively. The final ruleset came out to a total of 18 defined patterns and 3 generic path rules. The full list of rules can be seen in Appendix II.

Up to this point, the process was mostly based on UD representations of the CrowdTruth Cause dataset. For a next step, and due to the before-noted difficulties with grammatical representations, we attempted to generate an additional set of rules for AMR graphs, to evaluate the potential of combining representation systems into one complete classification model. During this attempt at creating AMR rules we realized that many of the respective graphs could not actually be matched due to issues with the conversion system, as outlined in more detail in section 2.3.4.

All in all the process gave us a good overview of the CrowdTruth Cause dataset, made us aware of POTATO's technical limitations in creating the type of classification rules we wanted, and provided us with a baseline classification score to compare against. Many of the suggestions and learnings taken from this process are outlined in section 3.2.

## 4.2 Advanced Entity Tagging

When investigating the new entity tagging features (as outlined in section 3.2.3), there were two key points we were interested in:

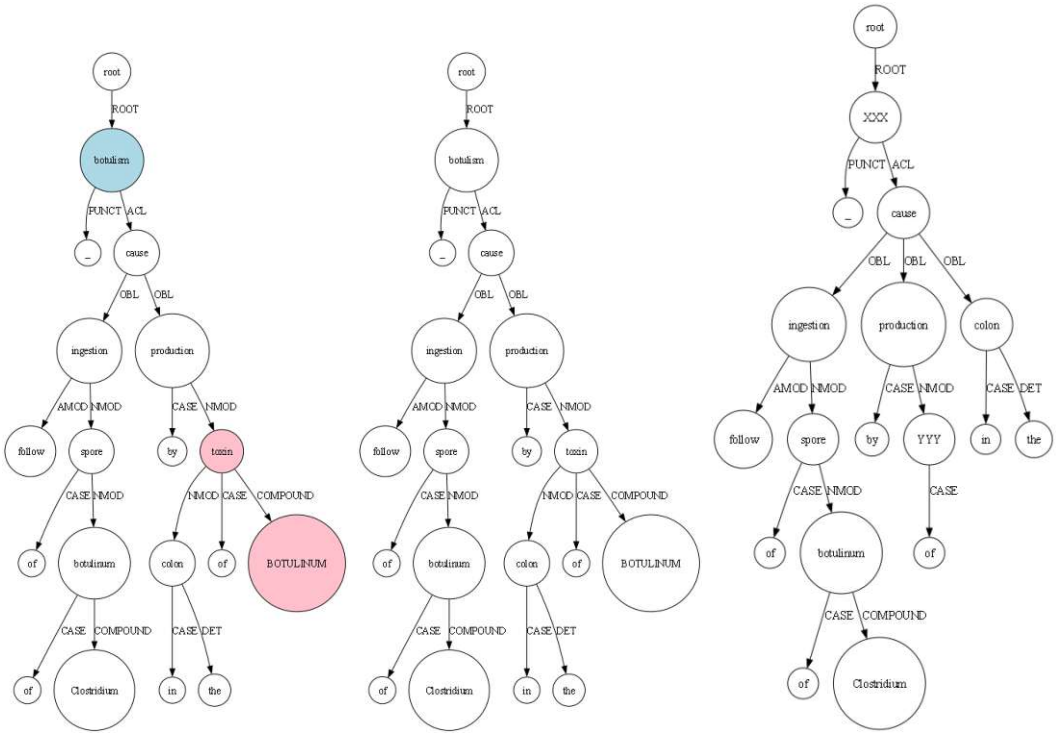
### 4.2.1 Fixing the conversion algorithm

At the outset we proposed that replacing various keywords with meaningless placeholder text might reduce the capability of a given conversion system to accurately model speech in its respective graph format. To illustrate this, refer to figures 4.3,4.4 and 4.5, which offer direct comparisons of the same graph converted with the old and new entity tagging systems across all representation systems. It is plain to see that the old version, using token replacements, deviates from the original sentence structure, while the new entity tagging system preserves it.

To give an example of how this can affect predictive quality, take a look at the rules in figure 4.6. We can observe that the previous rule in figure 4.6a and the new version in figure 4.6b are structurally equal. However, the new rule returns 9/10 correct hits on the train set. Compared to 7/9 using the old system, this is a very slight increase in both precision and recall.

While the change for this single rule is rather negligible, the idea is that these small performance increases should accumulate to a better performance, and allow for writing rules more accurately as a whole.

However, one problem still remains, which is that the algorithm heavily relies on accurate tagging in the source set, as well as a reliable way to align graph nodes with their original tokens. In the case of the CrowdTruth Cause dataset, we have noticed that some isolated




(a) New tagging (b) Source text (c) Old tagging

Figure 4.3: Comparing tagging systems to original graph structure (UD)
Sentence #1472: "BOTULISM caused by production of BOTULINUM TOXIN in the colon following ingestion of spores of Clostridium botulinum."

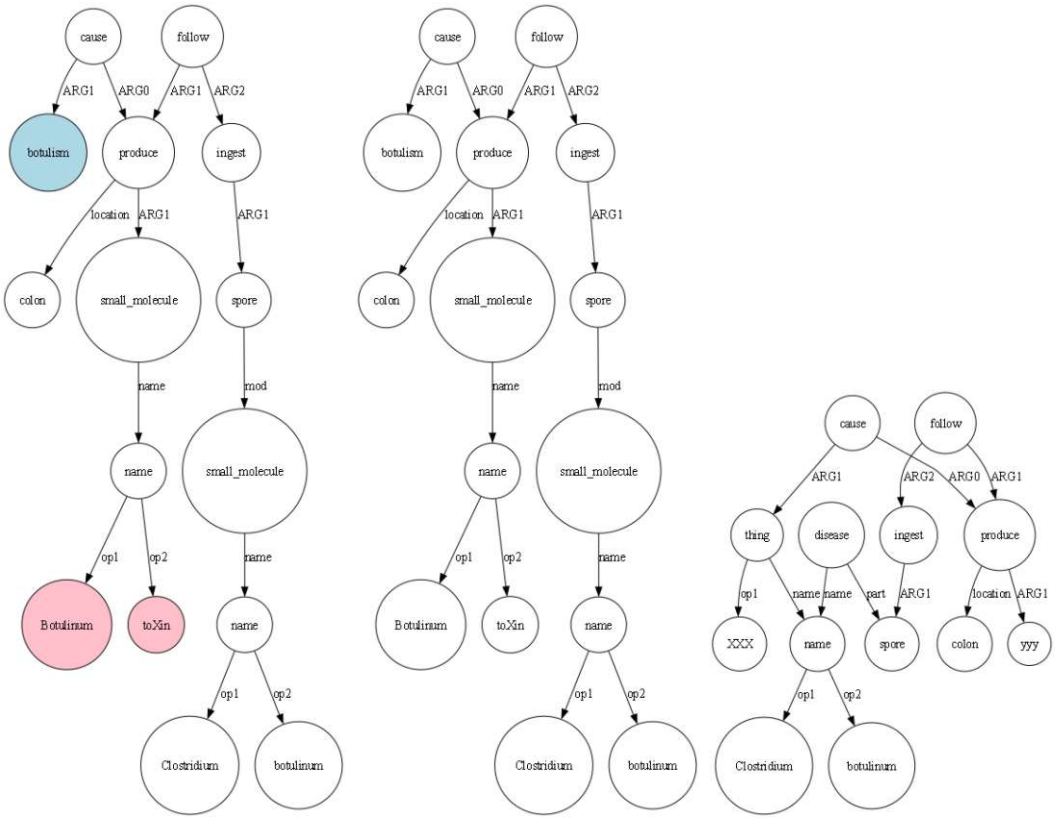
graphs were lost to the tagging system, due to inconsistencies with how the entities were marked in the source dataset. This depressed recall scores slightly.

In addition, it should be noted that the original algorithm replaced all occurrences of an entity, whereas the new tagging system tries to only target the specific token. This makes a difference in the case of repeat words. Compare for example figures 4.7 and 4.8. We can see that the old style of tagging produces multiple nodes with a 'YYY' label, which leads to a positive hit on UD graph #410 for rule 16. This doesn't work on the new system. This technically leads to a worse score, but a result that is more accurate to the rule definition. We argue that this encourages a more precise writing of rules.

For a full overview of performance gain on train, validation and test set using the new system, refer to table 5.1.

### 4.2.2 Use contents of entity nodes for classification

Perhaps the more significant change is that the new system allows a user to create rules that directly reference the content of an entity node. As outlined in the methods section,




(a) New tagging (b) Source text (c) Old tagging

Figure 4.4: Comparing tagging systems to original graph structure (AMR) Sentence #1472: "BOTULISM caused by production of BOTULINUM TOXIN in the colon following ingestion of spores of Clostridium botulinum."

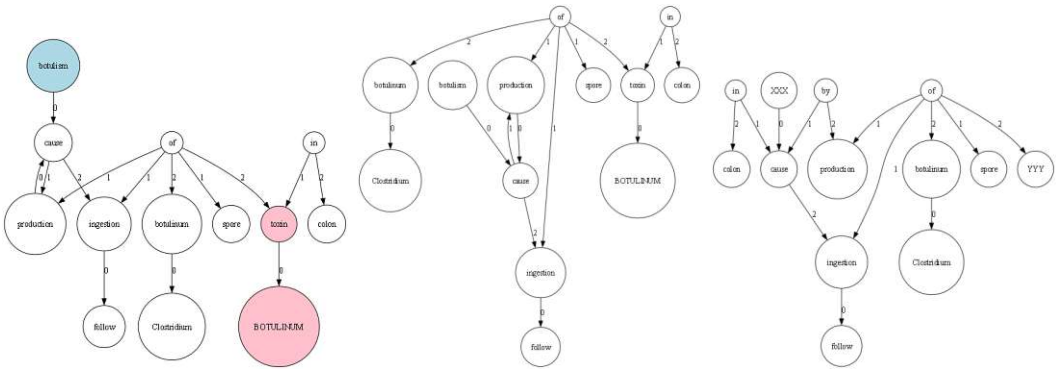
the hope is that this can be used to leverage domain knowledge.

Since medical cause-effect relationships are outside the scope of this work, we attempted to approximate such domain knowledge through statistical analysis. We have evaluated the most common individual entities as well as entity pairings. This allowed us to simulate two concepts in particular.

**1. If we are aware of common diseases and symptoms, we can use this knowledge to construct a rule**

Table 4.2 outlines the most common entity tokens found in positive-labeled observations in the CrowdTruth Cause dataset.

Now one thing we could do is to make use of our "simulated" medical knowledge and build a simple but effective rule that looks for the presence of common symptom and




(a) New tagging (b) Source text (c) Old tagging

Figure 4.5: Comparing tagging systems to original graph structure (FL).
Sentence #1472: "BOTULISM caused by production of BOTULINUM TOXIN in the colon following ingestion of spores of Clostridium botulinum."
*Note: New tagging and source are structurally identical but rendered differently*

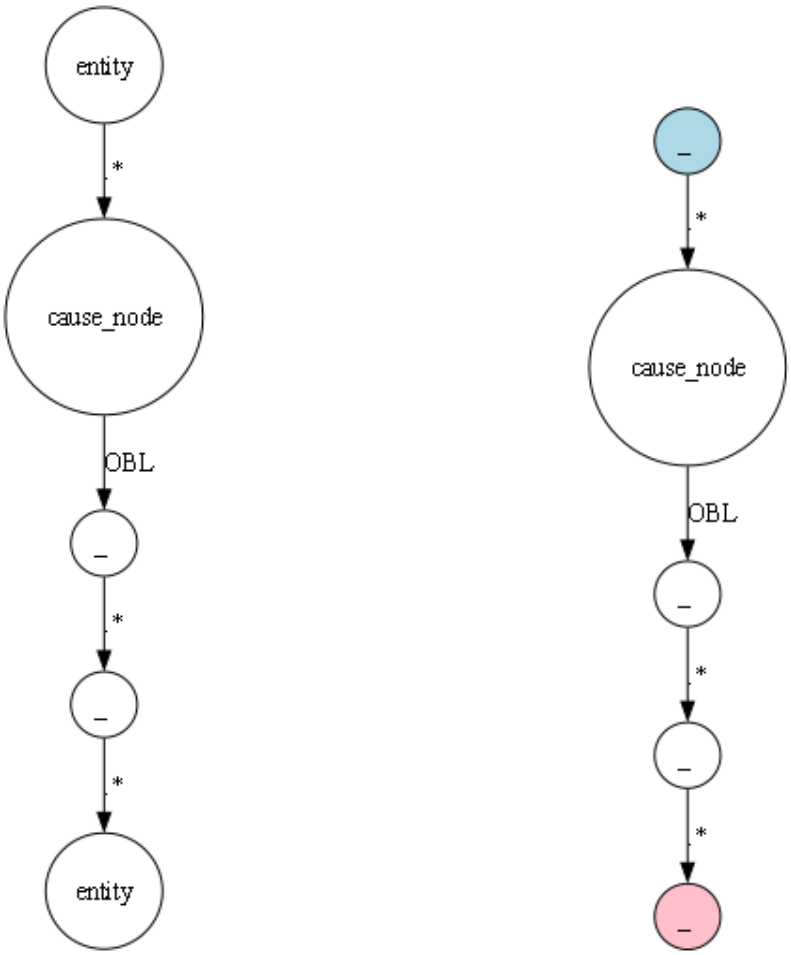
| Entity 1 | # | Entity 2 | # |
|---|---|---|---|
| PAIN | 86 | SYNDROME | 56 |
| SEIZURES | 27 | DISEASE | 38 |
| HYPERTENSION | 25 | VIRUS | 29 |
| DIARRHEA | 24 | CARCINOMA | 21 |
| FEVER | 20 | PHEOCHROMOCYTOMA | 20 |
| INFECTIONS | 17 | EPILEPSY | 16 |
| SYNDROME | 17 | DIABETES | 15 |
| HEADACHE | 16 | MONONUCLEOSIS | 15 |
| PREGNCY | 15 | CANCER | 15 |
| HEPATITIS | 12 | INFECTIOUS | 14 |
| BLEEDING | 12 | MIGRAINE | 13 |

Table 4.2: Most frequently occurring entities in the CrowdTruth Cause dataset

disease entities, such as this one:

```
(u_11 / PAIN|HYPERTENSION|SEIZURES|FEVER|DIARRHEA
    :entity (entity_0 / 1)
    ),
(u_13 / SYNDROME|DISEASE|CARCINOMA|VIRUS|
        PHEOCHROMOCYTOMA|EPILEPSY
    :entity (entity_1 / 2)
    )
```

This would give us a relatively good performance with 32/38 correct hits for 0.84 precision.



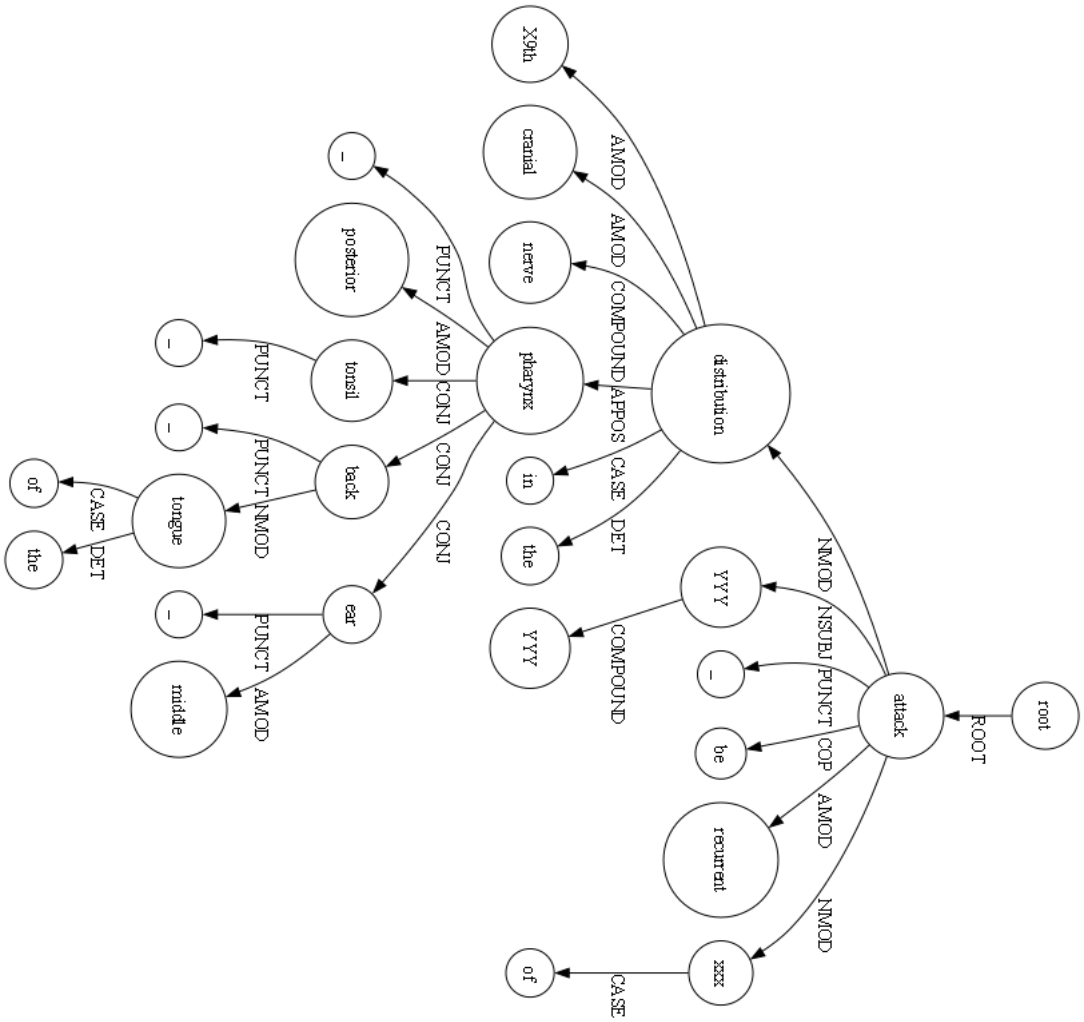

(a) Previous tagging system (b) New tagging system

Figure 4.6: Comparing ruleset pattern #3 before and after tagging system update. For visualization purposes, 'entity' and 'cause_node' are used as placeholder labels for RegEx disjunctions. The full penman notation can be seen in figures II.3 and III.3 respectively.

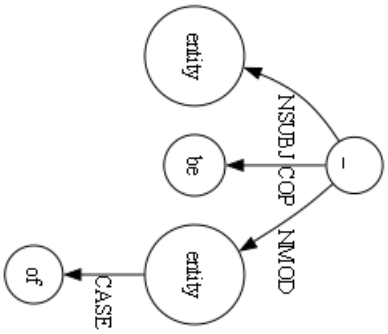
However, what are we really encoding here? That the co-occurrence of diseases and symptoms as entities makes it likely that a sentence is talking about a cause-effect relationship between them in a medical cause relation dataset. Take for example the false positive

#354 "Removal of PHEOCHROMOCYTOMA may also cure HYPERTENSION ( 77."

which does not necessarily talk about cause-effect explicitly. We could stick with the rule if we just wanted to increase coverage, but perhaps it is better to at least require the existence of a cause node as well:

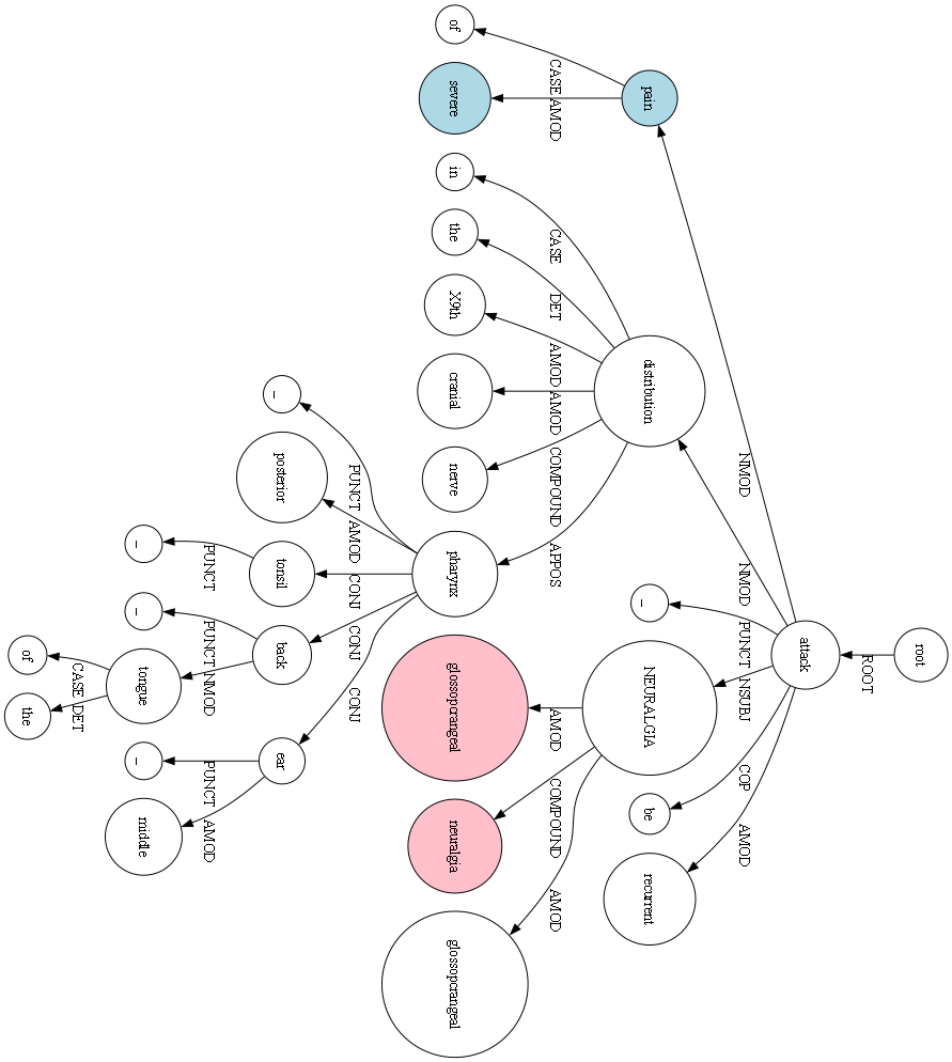



(a) UD representation: #410 - "GLOSSOPHARYNGEAL NEURALGIA GLOSSOPHARYNGEAL NEURALGIA is recurrent attacks of SEVERE PAIN in the 9th cranial nerve distribution (posterior pharynx, tonsils, back of the tongue, middle ear."

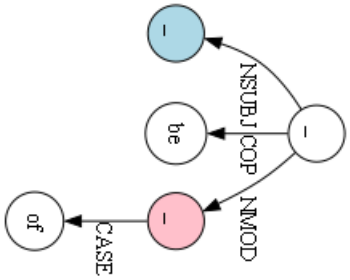



(b) Rule 16 will match

Figure 4.7: Matching rule 16 on the old system




(a) UD representation: #410 - "GLOSSOPHARYNGEAL NEURALGIA GLOSSOPHARYNGEAL NEURALGIA is recurrent attacks of SEVERE PAIN in the 9th cranial nerve distribution (posterior pharynx, tonsils, back of the tongue, middle ear."




(b) Rule 16 will no longer match

Figure 4.8: Matching rule 16 on the new system

```
(u_11 / PAIN|HYPERTENSION|SEIZURES|FEVER|DIARRHEA
    :entity (entity_0 / 1)
    ),
(u_13 / SYNDROME|DISEASE|CARCINOMA|VIRUS|
        PHEOCHROMOCYTOMA|EPILEPSY
    :entity (entity_1 / 2)
    ),
(u_4 / cause|result|occur|characterize|
       implicate|involve|reactivation|
       begin|suggest|exacerbate|begin|
       increase|precipitate|indicate|
       associate|experience)
```

This gives us a still very respectable 17/19 with a precision of 0.89 and a somewhat more meaningful match pattern.

Alternatively we could imagine a rule like this

```
5(
    (u_11 / PAIN|HYPERTENSION|SEIZURES|FEVER|DIARRHEA
        :entity (entity_0 / 1)
    ),
    (u_13 / SYNDROME|DISEASE|CARCINOMA|VIRUS|
            PHEOCHROMOCYTOMA|EPILEPSY
        :entity (entity_1 / 2)
    )
)

5(
    (u_13 / SYNDROME|DISEASE|CARCINOMA|VIRUS|
              PHEOCHROMOCYTOMA|EPILEPSY
        :entity (entity_1 / 2)
    ),
    (u_11 / PAIN|HYPERTENSION|SEIZURES|FEVER|DIARRHEA
        :entity (entity_0 / 1)
    )
)
```

which looks for known entities that refer to one another within 5 steps in a given language representation graph, indicating that they share a sub-sentence and directly relate to one another through grammatical concepts. This produces 11/11 hits with perfect precision.

| Entity 1 | Entity 2 | # |
|---|---|---|
| HYPERTENSION | PHEOCHROMOCYTOMA | 17 |
| PAIN | ENDOMETRIOSIS | 12 |
| HEADACHE | MIGRAINE | 11 |
| SEIZURES | EPILEPSY | 10 |
| PAIN | FIBROMYALGIA | 9 |

Table 4.3: Most common entity token co-occurrences in the CrowdTruth Cause set

**2. If we have access to knowledge that directly models this cause-effect relationship externally, we can create rules for these specific relationships.**

To approximate this, consider table 4.3 which is a collection of the most commonly occurring entity pairings in positive-labeled observations. This gives us a hint as to actual cause-effect relations between potential entities. This is particularly useful when building rules across otherwise ambivalent verb nodes, such as outlined in section 2.3.4. For instance, take sentence

> #2427 "Functiol imaging of PAIN in patients with PRIMARY FIBROMYALGIA"

A reference to "imaging of X in patients with Y" does not necessarily imply a cause-effect relationship. However, with the added context that PAIN is a known symptom of FIBROMYALGIA (which is the kind of knowledge a medical database would provide), we can craft a rule.

```
(u_1 / .*
    :.* (u_13 / PAIN
        :entity (entity_0 / 1)
        )
    :.* (u_13 / FIBROMYALGIA|ENDOMETRIOSIS
        :entity (entity_1 / 2)
        )
    :.* (u_2 / patient :.* (u_3 / in)
    )
)"
```

Which gives us two correct hits. This in itself is not a massive change to the overall classifier performance, but it demonstrates how, given that we could align with a medical database to extract these known cause-effect relationships dynamically (instead of hardcoding them into node labels), we can leverage entity node contents to build rules that bring some clarity into otherwise vague relations.

In summary, this new entity tagging system enables a range of rule writing capabilities that otherwise would not be possible. Even if a potential user would not have considered any of the above example rules useful to the cause-effect relation problem, the important takeaway is this: these experiments and considerations were not even possible with the previous entity tagging system.

Table 5.1 gives an overview of performance metrics for the final classifier with the inclusion of some rules using the above-stated entity-content logic. Appendix III provides a full listing of the new ruleset and individual rule performances across train and validation sets of the CrowdTruth Cause dataset.

# CHAPTER 5

# Results & Discussion

Attempting to build a ruleset on the CrowdTruth Treat dataset has helped us realize various challenges with the process. One key takeaway has been that building rules using the UD representation system can be somewhat counter-intuitive. Many times we would have assumed that a simple phrase, or more precisely its UD pattern, could be used to match common ideas across a wide range of graphs, only to find that the actual pattern would vary wildly between samples, depending on the grammatical context in which it occurred. This would often lead to us having to define patterns so specific that they turned meaningless and needed to be scrapped again. Perhaps these struggles make a good case for a combined use of representation systems in future work.

While some challenges were due to the nature of Universal Dependency graphs, there were still other problems that could be addressed by updating some part of the POTATO framework. Various rules could have been expressed more conveniently. Some ideas we could not express at all. These may have been lesser problems, but for what it is worth, they are all outlined in more detail in section 3.2.

On a positive note, the new tagging system has shown two specific types of improvement:

1. The graph representation of samples seems more accurate. While we have not confirmed for all 3990 graphs that the new conversion is now in line with the original structure, spot checks have been overwhelmingly positive, and rules have largely transitioned into the new tagging system either keeping their previous performance levels or slightly improving on them.

2. We have demonstrated that the content of entity nodes can be used to build sensible classification rules. We have neither domain-knowledge in the medical field nor is it within the scope of this work to test alignments to medical databases, but through a rough approximation to established medical knowledge (as portrayed by the dataset itself), we were able to build meaningful rules that made use of entity

node contents specifically and ended up contributing positively to all performance metrics.

The full CrowdTruth Cause performance metrics across various stages of changing the tagging system can be seen in table 5.1.

| | Prec | Recall | F1 | Acc |
|---|---|---|---|---|
| **Old tagging system** | | | | |
| Train | 0.63 | 0.33 | 0.43 | 0.67 |
| Val | 0.53 | 0.23 | 0.33 | 0.64 |
| Test | 0.47 | 0.23 | 0.31 | 0.70 |
| **New tagging system** | | | | |
| Train | 0.63 | 0.33 | 0.43 | 0.67 |
| Val | 0.54 | 0.27 | 0.36 | 0.64 |
| Test | 0.47 | 0.24 | 0.32 | 0.71 |
| **New tagging system with entity rules** | | | | |
| Train | 0.64 | 0.34 | 0.44 | 0.68 |
| Val | 0.55 | 0.28 | 0.37 | 0.65 |
| Test | 0.50 | 0.27 | 0.35 | 0.71 |

Table 5.1: Comparative metrics using the new entity tagging feature

There were some caveats to the new entity tagging system. One motivation that we outlined in section 2.3.4 was the missing of some entity tags in the converted graphs. This has not been conclusively fixed. We can see in table 5.2 that AMR as well as FL representations still miss a sizeable amount of observations.

| | FL | AMR |
|---|---|---|
| Train | 16.6% | 32.0% |
| Val | 17.0% | 33.8% |
| Test | 14.0% | 30.0% |

Table 5.2: Percentage of graphs with at least one missing entity

To some extent this is expected. FL conversion can be a reductive process, and it is hard to say what should happen if the conversion algorithm decides to remove an entity node. This question might be at the center of a future project. AMR conversion depends on a black-box model, which, as we have argued in section 2.2, can be unreliable and hard to debug. For future work, it would be one idea to look into a rule-based AMR converter that allows for conclusive node-to-token matching.

We now have a precedent in the system to introduce new node attributes and process them through the penman notation interface. Development might continue in this direction, enriching graph nodes with semantic and other meta-attributes. Perhaps then it could

also be a subject of experimentation to build rulesets through other means than penman notation strings.

One final suggestion for future enhancements is to fully leverage the new entity tagging system by establishing a streamlined process that allows users to align a converted graph dataset to an existing semantic repository. This could make it possible to utilize the full power of semantic modeling efforts across various disciplines of research in building more powerful linguistic models through external relations.

CHAPTER

# Appendix

## I Manual ruleset - verbalized rules for observations

Table I.1: Full list of positive match observations used in the initial phase of the manual ruleset creation process. Key sections are essentially the phrases to look for when establishing a RegEx or graph search pattern. The plain speech categorization is an effort to highlight common ideas connecting these individual grammatical expressions.

| ID | Sentence | Key section | Plain speech rule |
|---|---|---|---|
| 657 | Abstinence from YYY and food eating processes causes XXX in those with eating disorders. | Abstinence from YYY [...] causes XXX | direct cause |
| 1634 | Carpal Tunnel Syndrome affects the hands since it is an YYY that results in motor and XXX nerve. | YYY that results in [...] XXX | direct cause |
| 3165 | Treatment of dermatophyte infections of the toeil or fingeril (onychomycosis, XXX caused by susceptible YYY | XXX caused by [...] YYY | direct cause |
| 1025 | Treatment of XXX in severe YYY does not always result in clinical improvement in the patient's central nervous system. | Treatment of XXX in [...] YYY | implied symptom |
| 746 | XXX may result from abnormal epinephrine and norepinephrine YYY or from cortisol excess (ACTH secreting tumors. | XXX may result from [...] YYY | direct cause |
| 1116 | The Tat protein of the XXX has been implicated in the pathophysiology of the neurocognitive deficits associated with YYY | XXX has been implicated in [...] associated with YYY | implied symptom |

| | | | |
|---|---|---|---|
| 958 | Alysis of the mortality of children with YYY in Moscow in the eighties revealed a very high specific incidence of XXX the principal cause of lethal outcomes occurring in the period of the disease manifestation in more than a half of the alyzed cases. | Alysis of [...] YYY [...] revealed a very high specific incidence of XXX | implied symptom |
| 2140 | XXX Skin damage as a result of exposure to YYY | XXX [...] as a result of exposure to YYY | indirect cause |
| 2961 | YYY involves three types of XXX: a constant burning or deep aching; an intermittent spontaneous XXX with a jabbing or lanciting quality; and a superficial, sharp, or radiating XXX or itching provoked by light touch (allodynia), which is present in 90% of persons with YYY and often interferes with sleep. | YYY involves [...} XXX [...] XXX [...] is present in [...] persons with YYY | implied symptom |
| 1159 | 476 , 477 , 478 YYY Increased risk of reactivation of YYY including XXX (BKVN. | YYY including XXX | type of |
| 3154 | Loffler's syndrome (a subcategory of XXX with primary cardiac involvement), which occurs in the tropics, begins as an YYY followed by thrombus formation on the endocardium, chordae, and atrioventricular (AV) valves, progressing to fibrosis. | subcategory of XXX [...] begins as YYY | direct cause |
| 294 | Treatment of XXX + caused by YYY. | XXX caused by YYY | direct cause |
| 220 | Evidence: Treatment failure occurs in 9% of patients despite compliance with therapy, incomplete response occurs in 13% of treated patients, and XXX that is sufficiently severe to cause premature discontinuation of YYY occurs in 13% of patients ( 3 ; 152 ; 156. | XXX [...] sufficiently severe to cause [...] YYY | direct cause |
| 1608 | Some studies have found that YYY a common additive to oxymetazoline sal sprays, may damage sal epithelia and exacerbate XXX | YYY [...] may [...] exacerbate XXX | direct cause |
| 1897 | More chronic, cyclic pain, particularly XXX dyspareunia, and menorrhagia, suggests YYY or adenomyosis. | XXX dyspareunia [...] suggests YYY | implied symptom |
| 2876 | XXX are caused by YYY; at least 70 HPV types are linked to skin lesions. | XXX [...] caused by YYY | direct cause |

| | | | |
|---|---|---|---|
| 2691 | The authors report a case of a 13 year old boy who presented with XXX unsteadiness, diplopia and papilloedema due to YYY. | XXX unsteadiness [...] due to YYY. | direct cause |
| 3091 | A 74 year old woman presented with moderate YYY with diagnostic features of XXX | YYY with diagnostic features of XXX | direct cause |
| 615 | XXX is a potentially life-threatening complication in patients with YYY. | XXX is a [...] complication in patients with YYY. | implied symptom |
| 2954 | Smear negative XXX due to YYY acquired in the Amazon. | XXX due to YYY | direct cause |
| 433 | First, when YYY was first imported to the westernized continent from India, it was shown that this agent precipitated a XXX. | YYY [...] precipitated a XXX | indirect cause |
| 2333 | other symptoms in addition to XXX are also present, but there is clear symptom overlap among YYY, rhinosinusitis, and other sal passage/sinus pathologic conditions, and further research is needed | symptoms in addition to XXX [...] there is clear symptom overlap among YYY | implied symptom |
| 720 | Hyperlipidemia and XXX due to YYY and nolcoholic steatohepatitis may occur with lipodystrophy. | XXX due to YYY | direct cause |
| 3099 | XXX/facial pain or pressure of a dull, constant, or aching sort over the affected sinuses is common with both acute and chronic stages of YYY. | XXX [...] is common with [...] YYY. | indirect cause |
| 1447 | YYY occurs in 10 to 20% of patients; ovaries can become massively enlarged, and intravascular fluid volume shifts into the peritoneal space, causing potentially life threatening XXX and hypovolemia. | YYY occurs [...] causing [...] XXX | direct cause |
| 1330 | In this review, we discuss the clinical and histologic features of YYY a cutaneous disorder characterized by recurrent eruptions of self healing XXX and small nodules with histologic findings suggestive of malignt lymphoma. | YYY [...] characterized by [...] XXX | implied symptom |

| | | | |
|---|---|---|---|
| 195 | Other findings: tinea pedis (athlete's foot) and XXX are common Broad category of fungal il disease; may be caused by YYY yeasts, or molds Tinea unguium Major subtype of onychomycosis. | XXX [...] may be caused by YYY | direct cause |
| 1907 | YYY is a herpesvirus that causes XXX carcinomas and immunoproliferative disease. | YYY causes XXX | direct cause |
| 1290 | In investigation on carcinogenesis the first reports were published on the use of antisense oligonucleotides during inhibition of the development of tumours by a humoral mechanism and on the gene based YYY of the lungs, perhaps associated with the basis for the development of XXX | YYY [...] associated with [...] XXX | implied symptom |
| 785 | The YYY (FMS) is characterized by widespread XXX and diffuse tenderness in specified locations. | YYY is characterized by XXX | direct cause |
| 682 | Of the 2,585 men, 24 (0.93%) were positive for YYY indicating that XXX in the target group was below the intermediate endemicity. | YYY indicating [...] XXX | indirect cause |
| 759 | Treatment/Magement Children and Adults with YYY experience multiple XXX types that are resistant to most anti epileptic medications. | with YYY experience [...] XXX | direct cause |
| 410 | YYY YYY is recurrent attacks of XXX in the 9th cranial nerve distribution (posterior pharynx, tonsils, back of the tongue, middle ear. | YYY is [...] XXX | type of |
| 2804 | History Use of prescription, over the counter, and illicit drugs Anticoagulants, YYY or nonsteroidal anti inflammatory drugs, antiplatelet agents, and many prescription drugs may cause XXX | YYY [...] may cause XXX | direct cause |
| 2682 | XXX is the combition of Wilms' tumor (with WT1 deletion), aniridia, GU malformations (eg, rel hypoplasia, cystic disease, YYY | XXX is the combition of [...] YYY | type of |
| 735 | It is unknown why a patient with YYY suddenly develops a XXX, but the renin-angiotension system seems to play an important role | YYY [...] develops XXX | direct cause |

| | | | |
|---|---|---|---|
| 1881 | XXX XXX is neuromuscular poisoning from YYY | XXX is [...] from YYY | direct cause |
| 2021 | YYY from cranial nerve IX, causes XXX in the back of the throat or behind the angle of the jaw. | YYY [...] causes XXX | direct cause |
| 2477 | YYY is characterized by obesity, XXX retinitis pigmentosa, and polydactyly. | YYY is characterized by [...] XXX | implied symptom |
| 1872 | The elimition process can overcome XXX and unmask problem YYY so that the patients can associate cause and effect. | overcome XXX [...] unmask YYY | implied symptom |
| 956 | 120 YYY Infections Altertive to penicillin G for treatment of XXX caused by YYY perfringens or other YYY | XXX caused by YYY | direct cause |
| 2435 | African Americans generally tend to have a high risk of dying from a XXX, chiefly due to YYY and uncontrolled diabetes. | XXX [...] due to YYY | direct cause |
| 621 | This was because while carbamazepine and YYY are of roughly equal effectiveness, the former is less likely to cause sedation and XXX. | YYY [...] cause [...] XXX | direct cause |

# II Manual ruleset

This section contains an overview of all rules used in generating the original manual classification model using the old tagging system. For an overview of rule-wise performance on train and validation sets, refer to table II.1

Note: for better visualization, the graphs display placeholder text instead of the regular expressions used for cause verbs or entities. The actual content of a given graph is indicated in the penman notation

**Rule 1**

Example sentence: #657 - "Abstinence from YYY and food eating processes causes XXX in those with eating disorders."

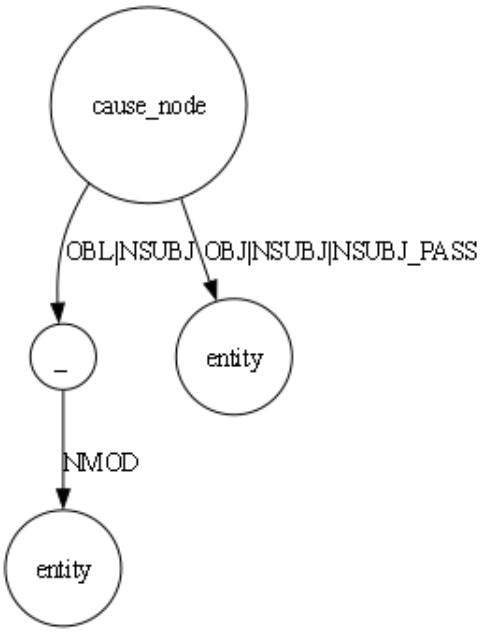



Figure II.1: Pattern rule 1

```
(u_1 / .*
    :NMOD (u_3 / XXX|YYY|xxj|yyj)
    :OBL|NSUBJ-of (u_4 / cause|result|occur|characterize|
                         implicate|involve|reactivation|
                         begin|suggest|exacerbate|begin|
                         increase|precipitate|indicate|
                         associate|experience
        :OBJ|NSUBJ|NSUBJ_PASS (u_5 / XXX|YYY|xxj|yyj)
    )
)
```

**Rule 2**

Example sentence: #785 - "The YYY (FMS) is characterized by widespread XXX and diffuse tenderness in specified locations."

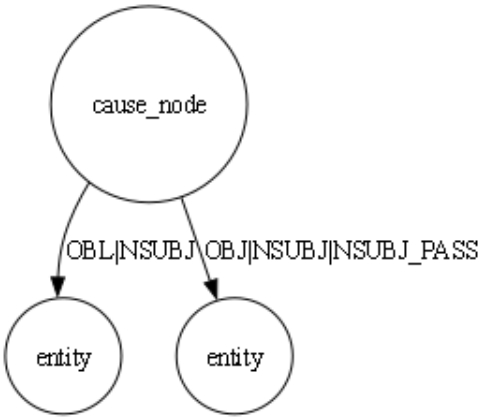



Figure II.2: Pattern rule 2

```
(u_4 / cause|result|occur|characterize|
       implicate|involve|reactivation|
       begin|suggest|exacerbate|begin|
       increase|precipitate|indicate|
       associate|experience
    :OBJ|NSUBJ|NSUBJ_PASS (u_5 / XXX|YYY|xxj|yyj)
    :OBL|NSUBJ (u_3 / XXX|YYY|xxj|yyj)
)
```

**Rule 3**

Example sentence: #1634 - "Carpal Tunnel Syndrome affects the hands since it is an YYY that results in motor and XXX nerve."

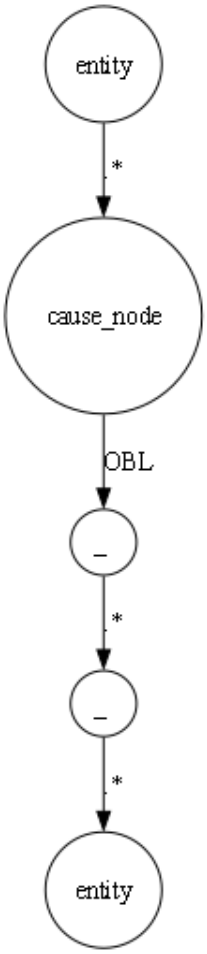



Figure II.3: Pattern rule 3

```
(u_11 / XXX|YYY|xxj|yyj
    :.* (u_13 / cause|result|occur|characterize|
                implicate|involve|reactivation|
                begin|suggest|exacerbate|begin|
                increase|precipitate|indicate|
                associate|experience
        :OBL (u_18 / .*
            :.* (u_15 / .*
                :.* (u_17 / XXX|YYY|xxj|yyj)
            )
        )
    )
)
```

**Rule 4**

Example sentence: #958 - "Alysis of the mortality of children with YYY in Moscow in the eighties revealed a very high specific incidence of XXX the principal cause of lethal outcomes occurring in the period of the disease manifestation in more than a half of the alyzed cases."

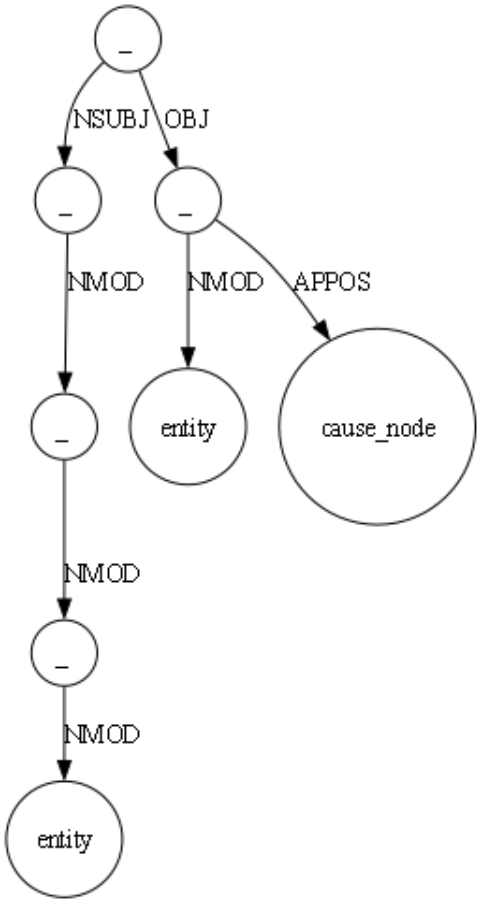



Figure II.4: Pattern rule 4

```
(u_1 / .*
    :NMOD (u_4 / .*
        :NMOD (u_6 / .*
            :NMOD (u_8 / XXX|YYY|xxj|yyj)
        )
    )
    :NSUBJ-of (u_14 / .*
        :OBJ (u_19 / .*
            :NMOD (u_21 / XXX|YYY|xxj|yyj)
            :APPOS (u_24 / cause|result|occur|characterize|
                           implicate|involve|reactivation|
                           begin|suggest|exacerbate|begin|
                           increase|precipitate|indicate|
                           associate|experience)
        )
    )
)
```

**Rule 5**

Example sentence: #2639 - "YYY is a gram negative bacillus that is the causative agent of XXX"

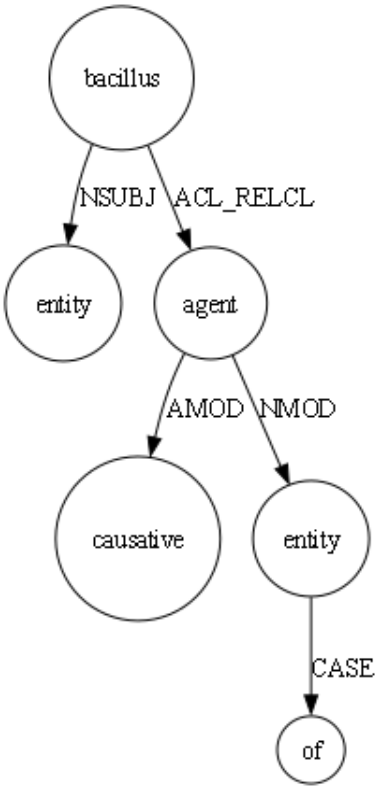



Figure II.5: Pattern rule 5

```
(u_6 / bacillus
    :NSUBJ (u_1 / XXX|YYY|xxj|yyj)
    :ACL_RELCL (u_11 / agent
        :AMOD (u_10 / causative)
        :NMOD (u_13 / XXX|YYY|xxj|yyj
            :CASE (u_12 / of)
        )
    )
)
```

**Rule 6**

Example sentence: #1872 - "The elimition process can overcome XXX and unmask problem YYY so that the patients can associate cause and effect."

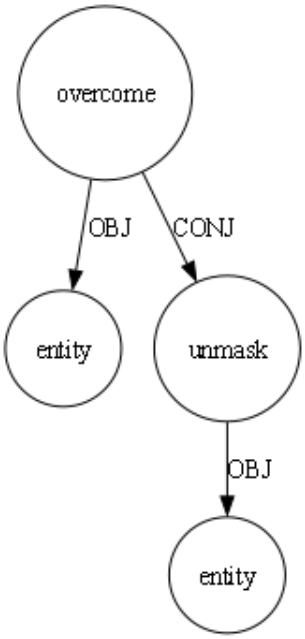



Figure II.6: Pattern rule 6

```
(u_8 / unmask
    :OBJ (u_10 / XXX|YYY|xxj|yyj)
    :CONJ-of (u_5 / overcome
        :OBJ (u_6 / XXX|YYY|xxj|yyj)
    )
)
```

**Rule 7**

Example sentence: #1473 - "Individuals already diagnosed with YYY or osteoporosis should discuss their exercise program with their physician to avoid XXXs."

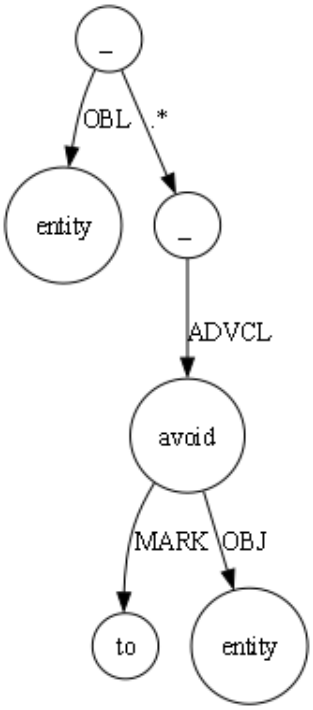



Figure II.7: Pattern rule 7

```
(u_3 / .*
    :OBL (u_5 / XXX|YYY|xxj|yyj)
    :.* (u_9 / .*
        :ADVCL (u_17 / avoid
            :MARK (u_16 / to)
                :OBJ (u_18 / XXX|YYY|xxj|yyj)
        )
    )
)
```

**Rule 8**

Example sentence: #2691 - "The authors report a case of a 13 year old boy who presented with XXX unsteadiness, diplopia and papilloedema due to YYY."

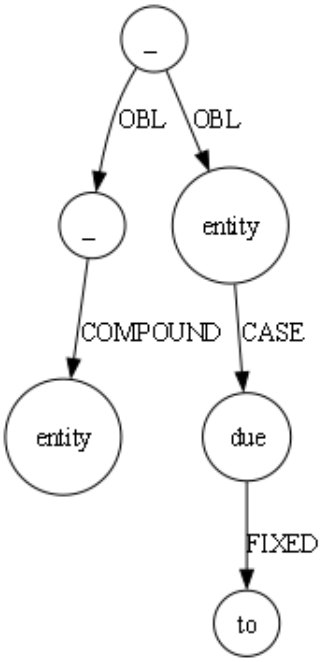



Figure II.8: Pattern rule 8

```
(u_13 / .*
    :OBL (u_16 / .*
        :COMPOUND (u_15 / XXX|YYY|xxj|yyj)
    )
    :OBL (u_23 / XXX|YYY|xxj|yyj
        :CASE (u_21 / due
            :FIXED (u_22 / to)
        )
    )
)
```

**Rule 9**

Example sentence: #995 - "Two other major causes of death include: hepatitis infections causing YYY and, obstruction of air or blood flow due to XXX."

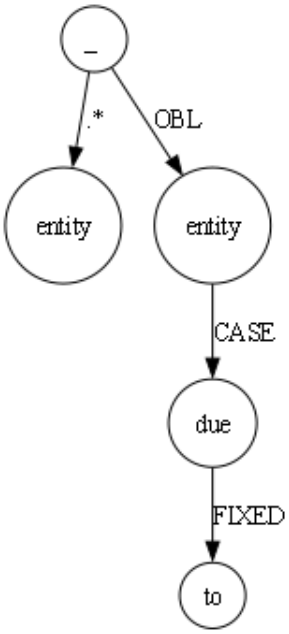



Figure II.9: Pattern rule 9

```
(u_13 / .*
    :OBL (u_23 / XXX|YYY|xxj|yyj
        :CASE (u_21 / due
            :FIXED (u_22 / to)
        )
    )
    :.* (u_15 / XXX|YYY|xxj|yyj)
)
```

**Rule 10**

Example sentence: #327 - "XXX due to YYY in congested Peyer's patches; this can be very serious but is usually not fatal."

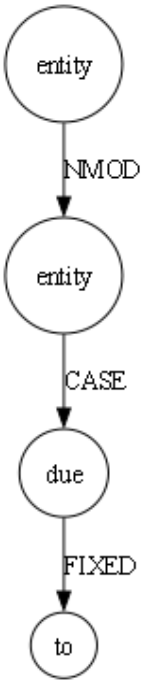



Figure II.10: Pattern rule 10

```
(u_1 / XXX|YYY|xxj|yyj
    :NMOD (u_4 / XXX|YYY|xxj|yyj
        :CASE (u_2 / due
            :FIXED (u_3 / to)
        )
    )
)
```

**Rule 11**

Example sentence: #615 - "XXX is a potentially life-threatening complication in patients with YYY."

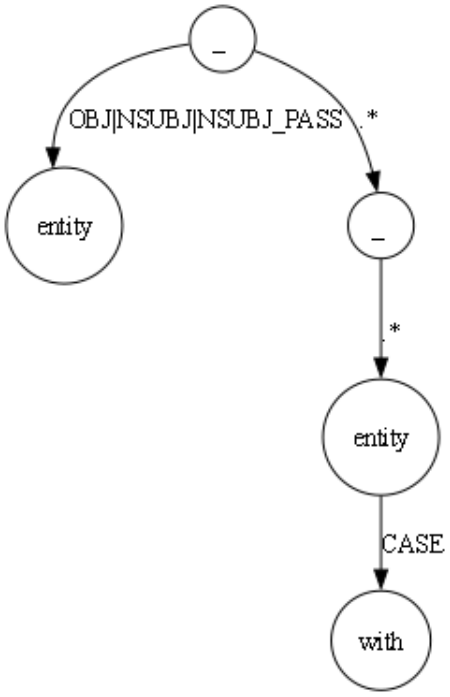



Figure II.11: Pattern rule 11

```
(u_6 / .*
    :OBJ|NSUBJ|NSUBJ_PASS (u_1 / XXX|YYY|xxj|yyj)
    :.* (u_8 / .*
        :.* (u_10 / XXX|YYY|xxj|yyj
            :CASE (u_9 / with)
        )
    )
)
```

**Rule 12**

Example sentence: #4 - "XXX appearing after ingesting potent YYY strongly predicts imminent serotonin toxicity ."

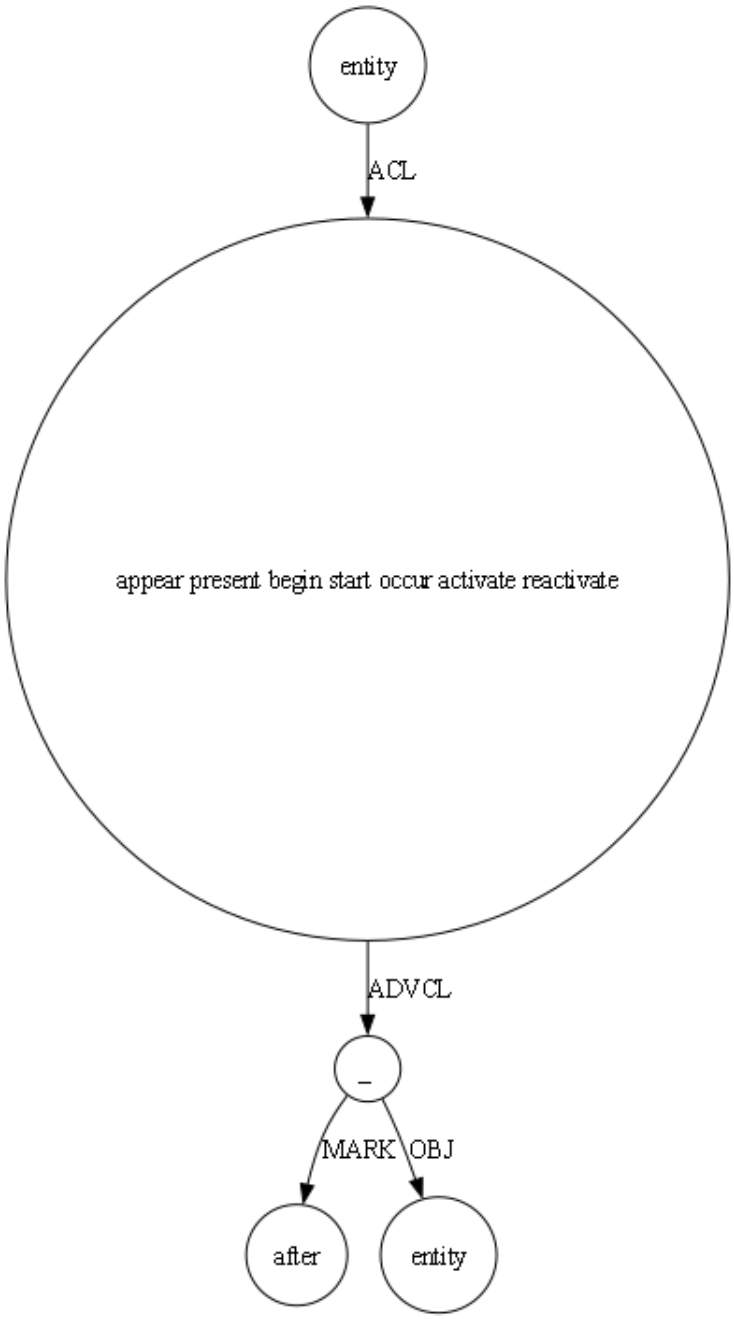



Figure II.12: Pattern rule 12

```
(u_1 / XXX|YYY|xxj|yyj
    :ACL (u_2 / appear|present|begin|start|occur|activate|reactivate
        :ADVCL (u_4 / .*
            :MARK (u_3 / after)
            :OBJ (u_6 / XXX|YYY|xxj|yyj)
        )
    )
)
```

**Rule 13**

Example sentence: #3099 - "XXX/facial pain or pressure of a dull, constant, or aching sort over the affected sinuses is common with both acute and chronic stages of YYY."

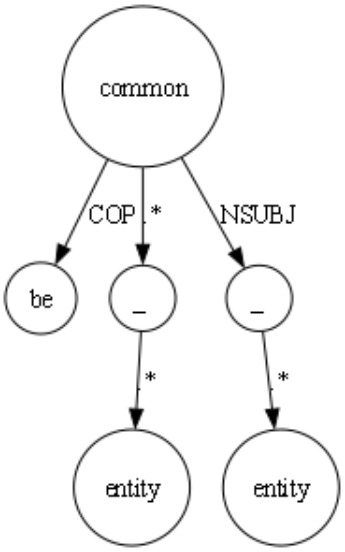



Figure II.13: Pattern rule 13

```
(u_4 / .*
    :.* (u_1 / XXX|YYY|xxj|yyj)
    :NSUBJ-of (u_21 / common
        :COP (u_20 / be)
        :.* (u_27 / .*
            :.* (u_29 / XXX|YYY|xxj|yyj)
        )
    )
)
```

**Rule 14**

Example sentence: #3091 - "A 74 year old woman presented with moderate YYY with diagnostic features of XXX"

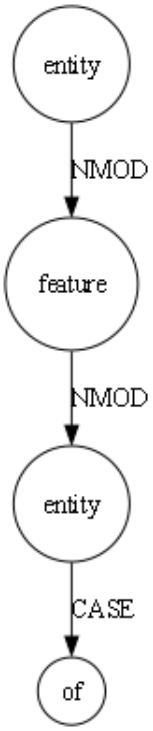



Figure II.14: Pattern rule 14

```
(u_9 / XXX|YYY|xxj|yyj
    :NMOD (u_12 / feature
        :NMOD (u_14 / XXX|YYY|xxj|yyj
            :CASE (u_13 / of)
        )
    )
)
```

**Rule 15**

Example sentence: #849 - "Other features of the YYY include ipsilateral congenital glaucoma anda XXX caused by leptomeningeal angiomatosis."

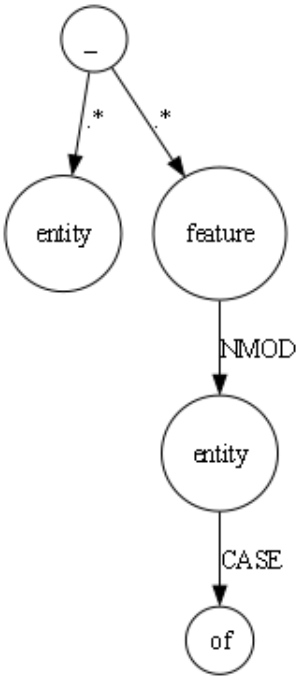



Figure II.15: Pattern rule 15

```
(u_2 / feature
    :NMOD (u_5 / XXX|YYY|xxj|yyj
        :CASE (u_3 / of)
    )
    :.*-of (u_6 / .*
        :.* (u_11 / XXX|YYY|xxj|yyj)
    )
)
```

**Rule 16**

Example sentence: #410 - "YYY YYY is recurrent attacks of XXX in the 9th cranial nerve distribution (posterior pharynx, tonsils, back of the tongue, middle ear."

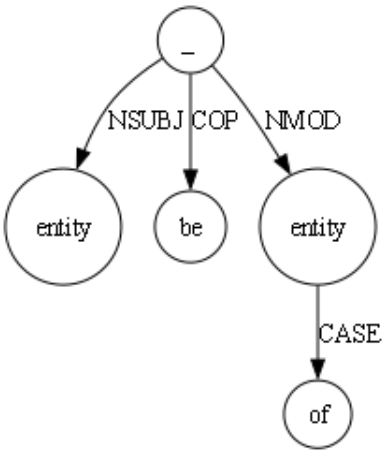



Figure II.16: Pattern rule 16

```
(u_5 / .*
    :NSUBJ (u_2 / XXX|YYY|xxj|yyj)
    :COP (u_3 / be)
    :NMOD (u_7 / XXX|YYY|xxj|yyj
        :CASE (u_6 / of)
    )
)
```

**Rule 17**

Example sentence: #1651 - "1 Drug and Alcohol Dependence Possible increased frequency of XXX and dependence in patients dependent on other YYY or alcohol; use with caution."

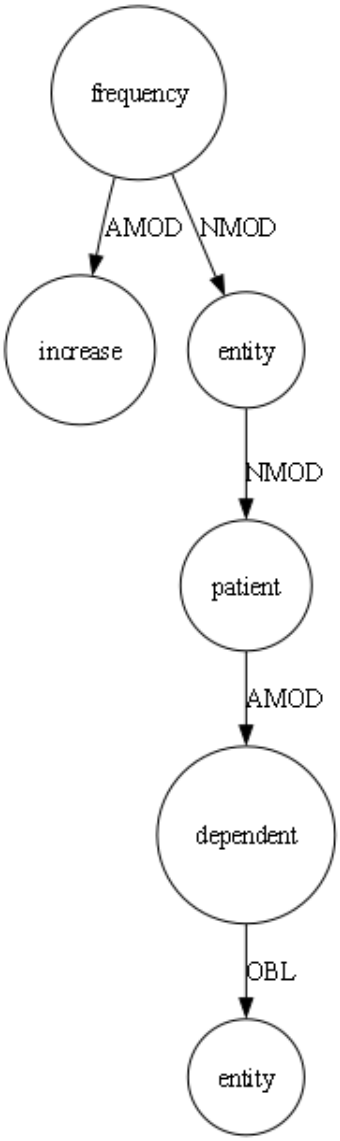



Figure II.17: Pattern rule 17

```
(u_7 / frequency
    :AMOD (u_6 / increase)
    :NMOD (u_9 / XXX|YYY|xxj|yyj
        :NMOD (u_13 / patient
            :AMOD (u_14 / dependent
                :OBL (u_17 / XXX|YYY|xxj|yyj)
            )
        )
    )
)
```

**Rule 18**

Example sentence: #1881 - "XXX XXX is neuromuscular poisoning from YYY"

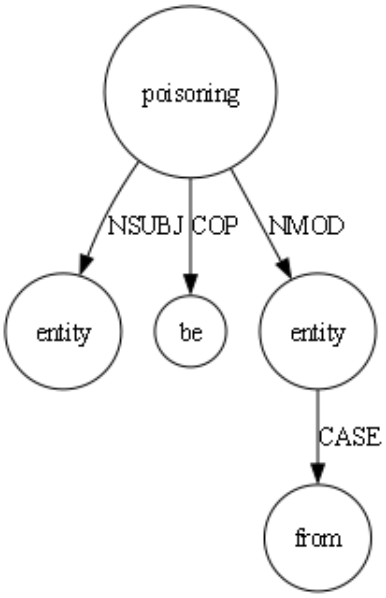



Figure II.18: Pattern rule 18

```
(u_5 / poisoning
    :NSUBJ (u_1 / XXX|YYY|xxj|yyj)
    :COP (u_3 / be)
    :NMOD (u_7 / XXX|YYY|xxj|yyj
        :CASE (u_6 / from)
    )
)
```

**Rule 19**
Example sentence: #774 - "fever with rel syndrome An ARTHROPOD BORNE VIRAL DISEASE caused by Hanta virus or related VIRUSES"

```
[
 (u_11 / YYY|yyj
    : (u_13 / cause|result|occur|characterize|
              implicate|involve|reactivation|
              begin|suggest|exacerbate|begin|
              increase|precipitate|indicate|
              associate|experience)
 ),
 path(
    (u_13 / cause|result|occur|characterize|
            implicate|involve|reactivation|
            begin|suggest|exacerbate|begin|
            increase|precipitate|indicate|
            associate|experience ),
    (u_17 / XXX|xxj)
 )
]
```

**Rule 20**
Example sentence: #1188 - "ECLAMPSIA defined by GENERALIZED SEIZURES may first occur following delivery simulating TTP ( 71."

```
[
 (u_13 / cause|result|occur|characterize|
         implicate|involve|reactivation|
         begin|suggest|exacerbate|begin|
         increase|precipitate|indicate|
         associate|experience
    : (u_11 / YYY|yyj)
 ),
 path(
    (u_13 / cause|result|occur|characterize|
            implicate|involve|reactivation|
            begin|suggest|exacerbate|begin|
            increase|precipitate|indicate|
            associate|experience ),
    (u_17 / XXX|xxj)
 )
]
```

**Rule 21**
Example sentence: #360 - "Surgery and/or treatment for prostate, colon and TESTICULAR CANCERS may result in SECONDARY LYMPHEDEMA particularly when lymph nodes have been removed or damaged."

```
[
 path(
    (u_13 / cause|result|occur|characterize|
            implicate|involve|reactivation|
            begin|suggest|exacerbate|begin|
            increase|precipitate|indicate|
            associate|experience),
    (u_11 / XXX|xxj)
 ),
 path(
    (u_13 / cause|result|occur|characterize|
            implicate|involve|reactivation|
            begin|suggest|exacerbate|begin|
            increase|precipitate|indicate|
            associate|experience ),
    (u_17 / YYY|yyj)
 )
]
```

**Negative rule 1**

Example sentence: #2066 - "YYY does not cause XXX although pain may limit muscular effort."

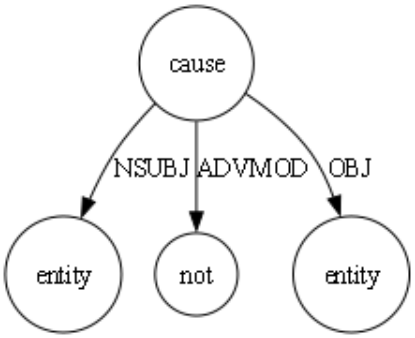



Figure II.19: Negative rule 1

```
(u_3 / cause
    :NSUBJ (u_1 / XXX|YYY|xxj|yyj)
    :ADVMOD (u_2 / not)
    :OBJ (u_4 / XXX|YYY|xxj|yyj)
)
```

| | Train | | | Val | | |
|---|---|---|---|---|---|---|
| Rule | Prec | Recall | F1 | Prec | Recall | F1 |
| 1 | 0.600000 | 0.012428 | 0.024351 | 0.500000 | 0.006711 | 0.013245 |
| 2 | 0.659574 | 0.025684 | 0.049442 | 0.714286 | 0.033557 | 0.064103 |
| 3 | 0.777778 | 0.005800 | 0.011513 | 0.000000 | 0.000000 | 0.000000 |
| 4 | 1.000000 | 0.000829 | 0.001656 | 0.000000 | 0.000000 | 0.000000 |
| 5 | 1.000000 | 0.000829 | 0.001656 | 0.000000 | 0.000000 | 0.000000 |
| 6 | 1.000000 | 0.000829 | 0.001656 | 0.000000 | 0.000000 | 0.000000 |
| 7 | 1.000000 | 0.000829 | 0.001656 | 0.000000 | 0.000000 | 0.000000 |
| 8 | 1.000000 | 0.000829 | 0.001656 | 0.000000 | 0.000000 | 0.000000 |
| 9 | 0.833333 | 0.004143 | 0.008244 | 0.000000 | 0.000000 | 0.000000 |
| 10 | 1.000000 | 0.000829 | 0.001656 | 0.000000 | 0.000000 | 0.000000 |
| 11 | 0.542857 | 0.015742 | 0.030596 | 0.333333 | 0.006711 | 0.013158 |
| 12 | 1.000000 | 0.000829 | 0.001656 | 0.000000 | 0.000000 | 0.000000 |
| 13 | 1.000000 | 0.000829 | 0.001656 | 0.000000 | 0.000000 | 0.000000 |
| 14 | 1.000000 | 0.000829 | 0.001656 | 0.000000 | 0.000000 | 0.000000 |
| 15 | 1.000000 | 0.000829 | 0.001656 | 0.000000 | 0.000000 | 0.000000 |
| 16 | 0.827586 | 0.019884 | 0.038835 | 0.666667 | 0.013423 | 0.026316 |
| 17 | 1.000000 | 0.000829 | 0.001656 | 0.000000 | 0.000000 | 0.000000 |
| 18 | 1.000000 | 0.000829 | 0.001656 | 0.000000 | 0.000000 | 0.000000 |
| 19 | 0.788136 | 0.077051 | 0.140377 | 0.750000 | 0.040268 | 0.076433 |
| 20 | 0.662116 | 0.160729 | 0.258667 | 0.586207 | 0.114094 | 0.191011 |
| 21 | 0.589520 | 0.223695 | 0.324324 | 0.509804 | 0.174497 | 0.260000 |

Table II.1: Metrics after initial ruleset creation

# III Ruleset with new entity tagging system

This section contains an overview of all rules used in generating the final classification model using the new tagging system and entity-content-aware rules. For an overview of rule-wise performance on train and validation sets, refer to table III.1

Note: for better visualization the graphs display placeholder text instead of the regular expressions used for cause verbs or entities. You can see the actual content of a given graph in the penman representation

**Rule 1**

Example sentence: #657 - "Abstinence from YYY and food eating processes causes XXX in those with eating disorders."

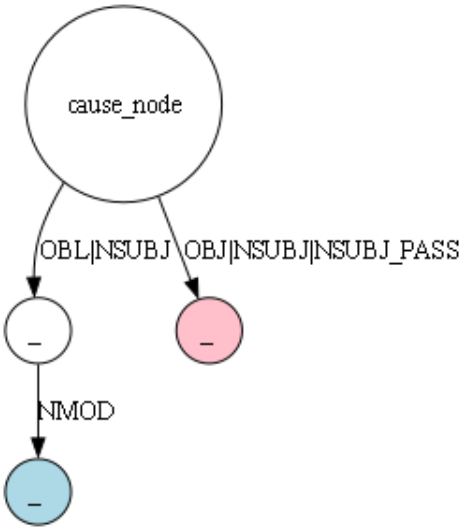



Figure III.1: New tagging - Pattern rule 1

```
(u_1 / .*
    :NMOD (u_3 / .*
        :entity (entity_0 / 1)
    )
    :OBL|NSUBJ-of (u_4 / cause|result|occur|characterize|
                         implicate|involve|reactivation|
                         begin|suggest|exacerbate|begin|
                         increase|precipitate|indicate|
                         associate|experience
        :OBJ|NSUBJ|NSUBJ_PASS (u_5 / .*
            :entity (entity_1 / 2)
        )
    )
)
```

**Rule 2**

Example sentence: #785 - "The YYY (FMS) is characterized by widespread XXX and diffuse tenderness in specified locations."

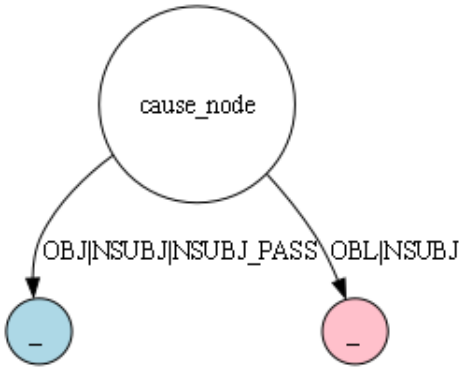



Figure III.2: New tagging - Pattern rule 2

```
(u_4 / cause|result|occur|characterize|
       implicate|involve|reactivation|
       begin|suggest|exacerbate|begin|
       increase|precipitate|indicate|
       associate|experience
    :OBJ|NSUBJ|NSUBJ_PASS (u_5 / .*
        :entity (entity_0 / 1)
    )
    :OBL|NSUBJ (u_3 / .*
        :entity (entity_1 / 2)
    )
)
```

**Rule 3**

Example sentence: #1634 - "Carpal Tunnel Syndrome affects the hands since it is an YYY that results in motor and XXX nerve."

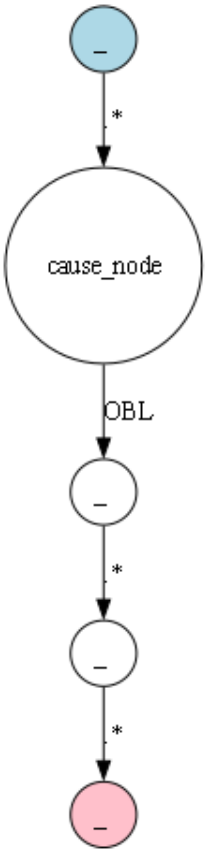



Figure III.3: New tagging - Pattern rule 3

```
(u_11 / .*
    :.* (u_13 / cause|result|occur|characterize|
                implicate|involve|reactivation|
                begin|suggest|exacerbate|begin|
                increase|precipitate|indicate|
                associate|experience
                    :OBL (u_18 / .*
                        :.* (u_15 / .*
                            :.* (u_17 / .*
                                :entity (entity_0 / 2)
                            )
                        )
                    )
    )
    :entity (entity_1 / 1)
)
```

**Rule 4**

Example sentence: #958 - "Alysis of the mortality of children with YYY in Moscow in the eighties revealed a very high specific incidence of XXX the principal cause of lethal outcomes occurring in the period of the disease manifestation in more than a half of the alyzed cases."

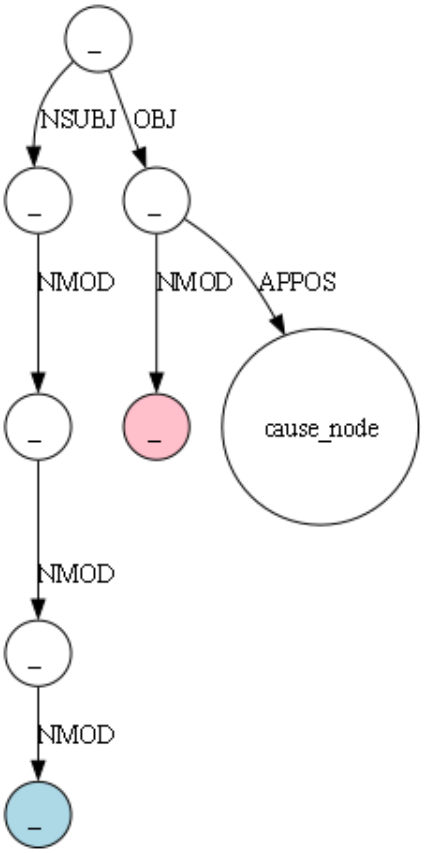



Figure III.4: New tagging - Pattern rule 4

```
(u_1 / .*
    :NMOD (u_4 / .*
        :NMOD (u_6 / .*
            :NMOD (u_8 / .*
                :entity (entity_0 / 1)
            )
        )
    )
    :NSUBJ-of (u_14 / .*
        :OBJ (u_19 / .*
            :NMOD (u_21 / .*
                :entity (entity_1 / 2)
            )
            :APPOS (u_24 / cause|result|occur|characterize|
                          implicate|involve|reactivation|
                          begin|suggest|exacerbate|begin|
                          increase|precipitate|indicate|
                          associate|experience)
        )
    )
)
```

**Rule 5**

Example sentence: #2639 - "YYY is a gram negative bacillus that is the causative agent of XXX"

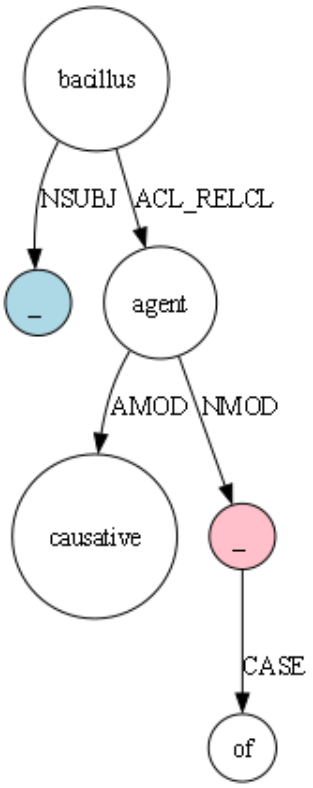



Figure III.5: New tagging - Pattern rule 5

```
(u_6 / bacillus
    :NSUBJ (u_1 / .*
        :entity (entity_0 / 1)
    )
    :ACL_RELCL (u_11 / agent
        :AMOD (u_10 / causative)
        :NMOD (u_13 / .*
            :entity (entity_1 / 2)
            :CASE (u_12 / of)
        )
    )
)
```

**Rule 6**

Example sentence: #1872 - "The elimition process can overcome XXX and unmask problem YYY so that the patients can associate cause and effect."

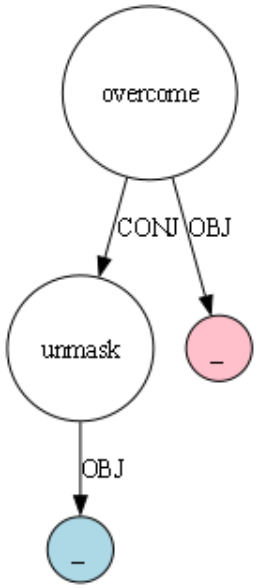



Figure III.6: New tagging - Pattern rule 6

```
(u_8 / unmask
    :OBJ (u_10 / .*
        :entity (entity_0 / 1)
    )
    :CONJ-of (u_5 / overcome
        :OBJ (u_6 / .*
            :entity (entity_1 / 2)
        )
    )
)
```

**Rule 7**

Example sentence: #1473 - "Individuals already diagnosed with YYY or osteoporosis should discuss their exercise program with their physician to avoid XXXs."

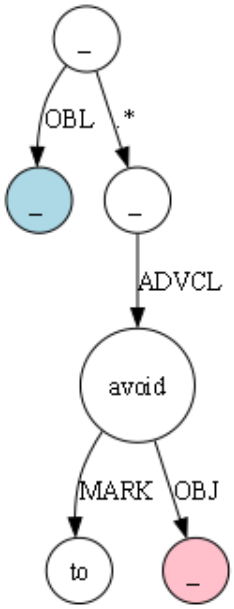



Figure III.7: New tagging - Pattern rule 7

```
(u_3 / .*
    :OBL (u_5 / .*
        :entity (entity_0 / 1)
    )
    :.* (u_9 / .*
        :ADVCL (u_17 / avoid
            :MARK (u_16 / to)
            :OBJ (u_18 / .*
                :entity (entity_1 / 2)
            )
        )
    )
)
```

**Rule 8**

Example sentence: #2691 - "The authors report a case of a 13 year old boy who presented with XXX unsteadiness, diplopia and papilloedema due to YYY."

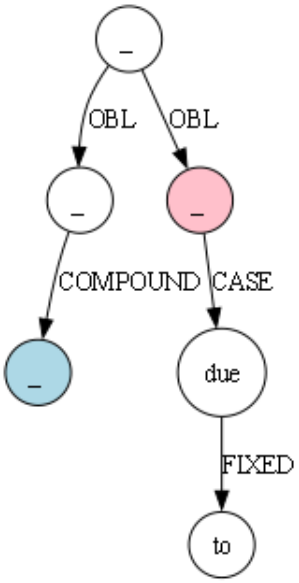



Figure III.8: New tagging - Pattern rule 8

```
(u_13 / .*
    :OBL (u_16 / .*
        :COMPOUND (u_15 / .*
            :entity (entity_0 / 1)
        )
    )
    :OBL (u_23 / .*
        :entity (entity_1 / 2)
        :CASE (u_21 / due
            :FIXED (u_22 / to)
        )
    )
)
```

**Rule 9**

Example sentence: #995 - "Two other major causes of death include: hepatitis infections causing YYY and, obstruction of air or blood flow due to XXX."

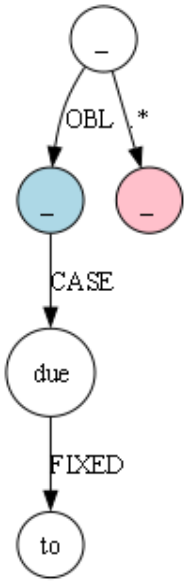



Figure III.9: New tagging - Pattern rule 9

```
(u_13 / .*
    :OBL (u_23 / .*
        :entity (entity_0 / 1)
        :CASE (u_21 / due
            :FIXED (u_22 / to)
        )
    )
    :.* (u_15 / .*
        :entity (entity_1 / 2)
    )
)
```

**Rule 10**

Example sentence: #327 - "XXX due to YYY in congested Peyer's patches; this can be very serious but is usually not fatal."

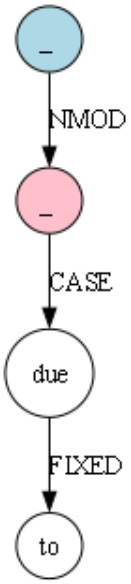



Figure III.10: New tagging - Pattern rule 10

```
(u_1 / .*
    :entity (entity_0 / 1)
    :NMOD (u_4 / .*
        :entity (entity_1 / 2)
        :CASE (u_2 / due
            :FIXED (u_3 / to)
        )
    )
)
```

**Rule 11**

Example sentence: #615 - "XXX is a potentially life-threatening complication in patients with YYY."

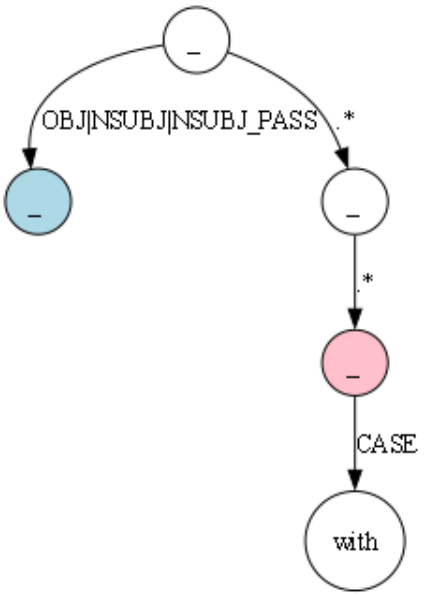



Figure III.11: New tagging - Pattern rule 11

```
(u_6 / .*
    :OBJ|NSUBJ|NSUBJ_PASS (u_1 / .*
        :entity (entity_0 / 1)
    )
    :.* (u_8 / .*
        :.* (u_10 / .*
            :entity (entity_1 / 2)
            :CASE (u_9 / with)
        )
    )
)
```

**Rule 12**

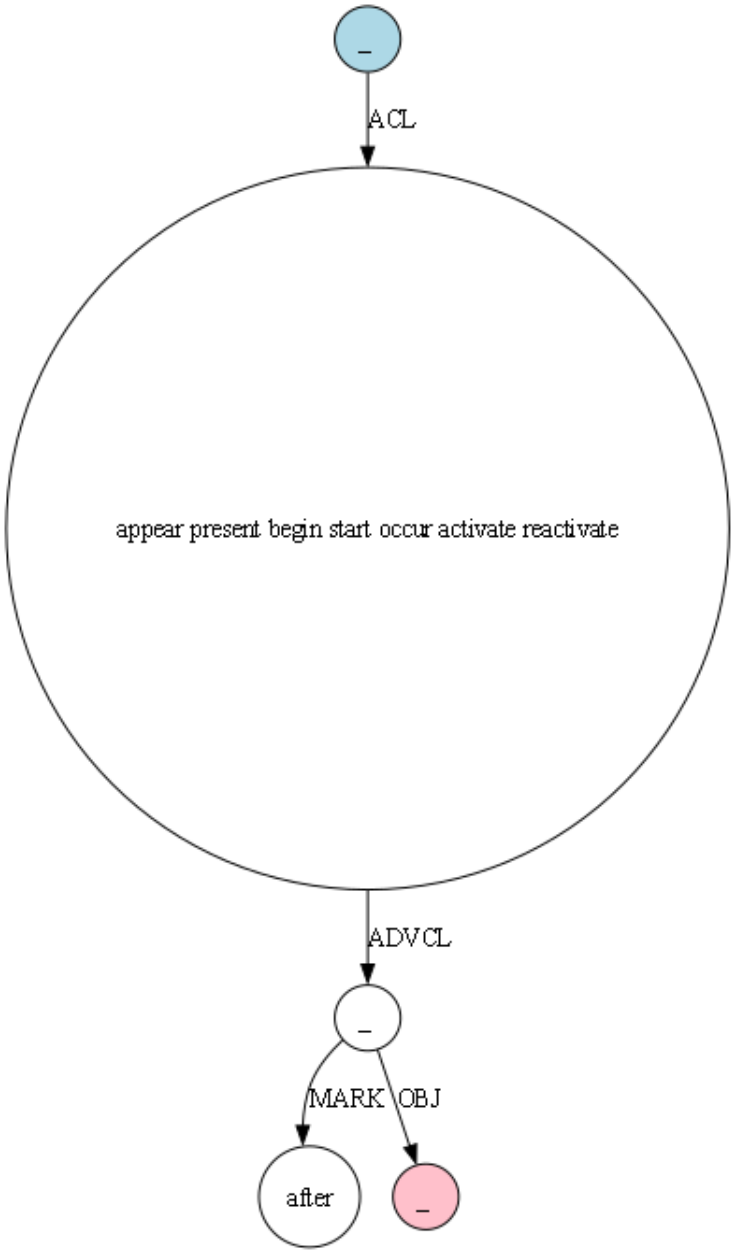



Figure III.12: New tagging - Pattern rule 12

```
(u_1 / .*
    :entity (entity_0 / 1)
    :ACL (u_2 / appear|present|begin|start|occur|activate|reactivate
        :ADVCL (u_4 / .*
            :MARK (u_3 / after)
            :OBJ (u_6 / .*
                :entity (entity_1 / 2)
            )
        )
    )
)
```

**Rule 13**

Example sentence: #3099 - "XXX/facial pain or pressure of a dull, constant, or aching sort over the affected sinuses is common with both acute and chronic stages of YYY."

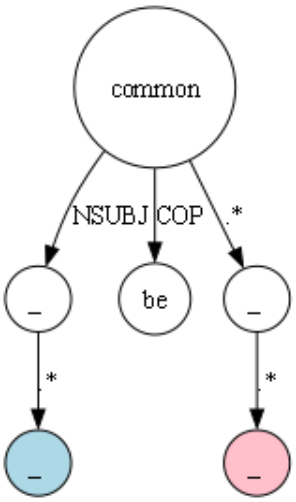



Figure III.13: New tagging - Pattern rule 13

```
(u_4 / .*
    :.* (u_1 / .*
        :entity (entity_0 / 1)
    )
    :NSUBJ-of (u_21 / common
        :COP (u_20 / be)
        :.* (u_27 / .*
            :.* (u_29 / .*
                :entity (entity_1 / 2)
            )
        )
    )
)
```

**Rule 14**

Example sentence: #3091 - "A 74 year old woman presented with moderate YYY with diagnostic features of XXX"

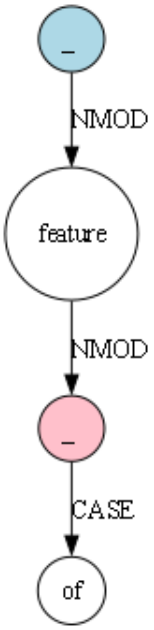



Figure III.14: New tagging - Pattern rule 14

```
(u_9 / .*
    :entity (entity_0 / 1)
    :NMOD (u_12 / feature
        :NMOD (u_14 / .*
            :entity (entity_1 / 2)
            :CASE (u_13 / of)
        )
    )
)
```

**Rule 15**

Example sentence: #849 - "Other features of the YYY include ipsilateral congenital glaucoma anda XXX caused by leptomeningeal angiomatosis."

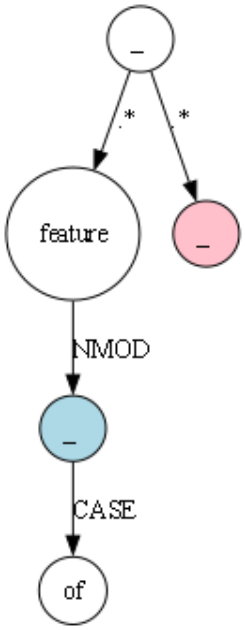



Figure III.15: New tagging - Pattern rule 15

```
(u_2 / feature
    :NMOD (u_5 / .*
        :entity (entity_0 / 1)
        :CASE (u_3 / of)
    )
    :.*-of (u_6 / .*
        :.* (u_11 / .*
            :entity (entity_1 / 2)
        )
    )
)
```

**Rule 16**

Example sentence: #410 - "YYY YYY is recurrent attacks of XXX in the 9th cranial nerve distribution (posterior pharynx, tonsils, back of the tongue, middle ear."

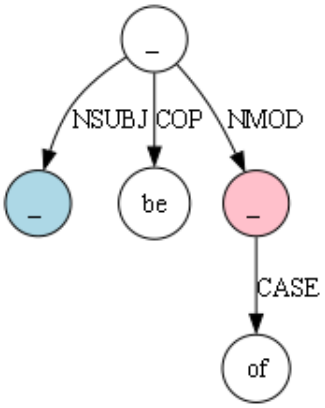



Figure III.16: New tagging - Pattern rule 16

```
(u_5 / .*
    :NSUBJ (u_2 / .*
        :entity (entity_0 / 1)
    )
    :COP (u_3 / be)
    :NMOD (u_7 / .*
        :entity (entity_1 / 2)
        :CASE (u_6 / of)
    )
)
```

**Rule 17**

Example sentence: #1651 - "1 Drug and Alcohol Dependence Possible increased frequency of XXX and dependence in patients dependent on other YYY or alcohol; use with caution."

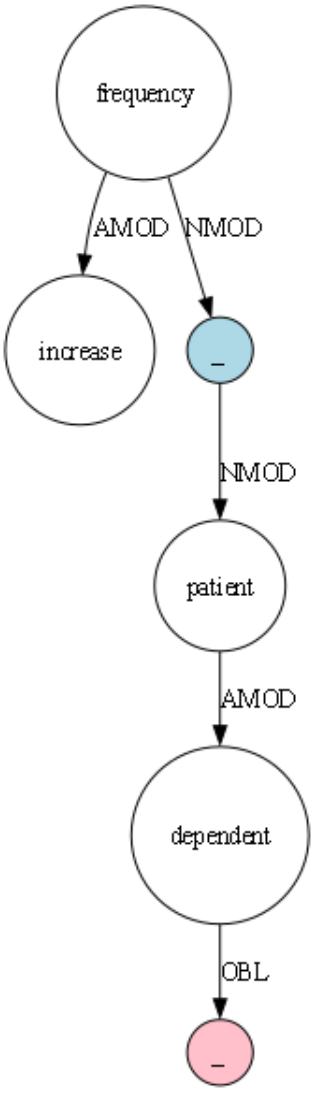



Figure III.17: New tagging - Pattern rule 17

```
(u_7 / frequency
    :AMOD (u_6 / increase)
    :NMOD (u_9 / .*
        :entity (entity_0 / 1)
        :NMOD (u_13 / patient
            :AMOD (u_14 / dependent
                :OBL (u_17 / .*
                    :entity (entity_1 / 2)
                )
            )
        )
    )
)
```

**Rule 18**

Example sentence: #1881 - "XXX XXX is neuromuscular poisoning from YYY"

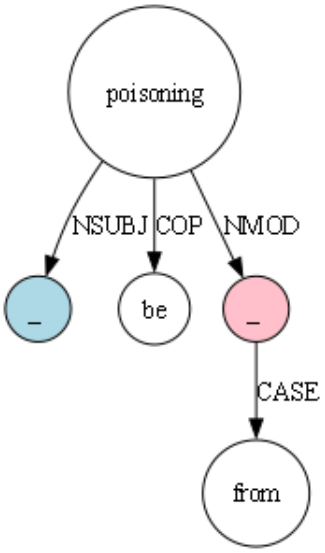



Figure III.18: New tagging - Pattern rule 18

```
(u_5 / poisoning
    :NSUBJ (u_1 / .*
        :entity (entity_0 / 1)
    )
    :COP (u_3 / be)
    :NMOD (u_7 / .*
        :entity (entity_1 / 2)
        :CASE (u_6 / from)
    )
)
```

**Rule 19**
Example sentence: #774 - "fever with rel syndrome An ARTHROPOD BORNE VIRAL DISEASE caused by Hanta virus or related VIRUSES"

```
(u_11 / .*
    :entity (entity_0 / 1)
    : (u_13 / cause|result|occur|characterize|
              implicate|involve|reactivation|
              begin|suggest|exacerbate|begin|
              increase|precipitate|indicate|
              associate|experience)
),
path(
    (u_13 / cause|result|occur|characterize|
            implicate|involve|reactivation|
            begin|suggest|exacerbate|begin|
            increase|precipitate|indicate|
            associate|experience ),
    (u_17 / .*
        :entity (entity_1 / 2)
    )
)
```

**Rule 20**
Example sentence: #1188 - "ECLAMPSIA defined by GENERALIZED SEIZURES may first occur following delivery simulating TTP ( 71."

```
(u_13 / cause|result|occur|characterize|
        implicate|involve|reactivation|
        begin|suggest|exacerbate|begin|
        increase|precipitate|indicate|
        associate|experience
    : (u_11 / .*
        :entity (entity_0 / 1)
    )
),
path(
    (u_13 / cause|result|occur|characterize|
            implicate|involve|reactivation|
            begin|suggest|exacerbate|begin|
            increase|precipitate|indicate|
```

```
            associate|experience ),
    (u_17 / .*
        :entity (entity_1 / 2)
    )
)
```

**Rule 21**
Example sentence: #360 - "Surgery and/or treatment for prostate, colon and TESTICULAR CANCERS may result in SECONDARY LYMPHEDEMA particularly when lymph nodes have been removed or damaged."

```
path(
    (u_13 / cause|result|occur|characterize|
            implicate|involve|reactivation|
            begin|suggest|exacerbate|begin|
            increase|precipitate|indicate|
            associate|experience),
    (u_11 / .*
        :entity (entity_0 / 1)
    )
),
path(
    (u_13 / cause|result|occur|characterize|
            implicate|involve|reactivation|
            begin|suggest|exacerbate|begin|
            increase|precipitate|indicate|
            associate|experience ),
    (u_17 / .*
        :entity (entity_1 / 2)
    )
)
```

**Rule 22**
Example sentence: #56 - "HYPERTENSION IN MEN IIA patients with PHEOCHROMOCYTOMA is more often paroxysmal than sustained, in contrast to the usual sporadic case."

```
5(
    (u_11 / PAIN|HYPERTENSION|SEIZURES|FEVER|
            DIARRHEA|HEADACHE|BLEEDING|INFECTIONS
        :entity (entity_0 / 1)
```

```
    ),
    (u_13 / SYNDROME|DISEASE|CARCINOMA|VIRUS|
            PHEOCHROMOCYTOMA|EPILEPSY|DIABETES|
            CANCER|MONONUCLEOSIS
        :entity (entity_1 / 2)
    )
)

5(
    (u_13 / SYNDROME|DISEASE|CARCINOMA|VIRUS|
            PHEOCHROMOCYTOMA|EPILEPSY|DIABETES|
            CANCER|MONONUCLEOSIS
        :entity (entity_1 / 2)
    ),
    (u_11 / PAIN|HYPERTENSION|SEIZURES|FEVER|
            DIARRHEA|HEADACHE|BLEEDING|INFECTIONS
        :entity (entity_0 / 1)
    )
)
```

**Rule 23**
Example sentence: #2294 - "METABOLIC SYNDROME, a combition of abdomil obesity, HYPERTENSION, insulin resistance and abnormal lipid levels occurs in nearly 75% of cases."

```
(u_11 / PAIN|HYPERTENSION|SEIZURES|FEVER|
        DIARRHEA|HEADACHE|BLEEDING|INFECTIONS
    :entity (entity_0 / 1)
),
(u_13 / SYNDROME|DISEASE|CARCINOMA|VIRUS|
        PHEOCHROMOCYTOMA|EPILEPSY|DIABETES|
        CANCER|MONONUCLEOSIS
    :entity (entity_1 / 2)
),
(u_4 / cause|result|occur|characterize|
       implicate|involve|reactivation|
       begin|suggest|exacerbate|begin|
       increase|precipitate|indicate|
       associate|experience
)
```

**Rule 24**

Example sentence: #2427 - "Functiol imaging of PAIN in patients with PRIMARY FIBROMYALGIA"

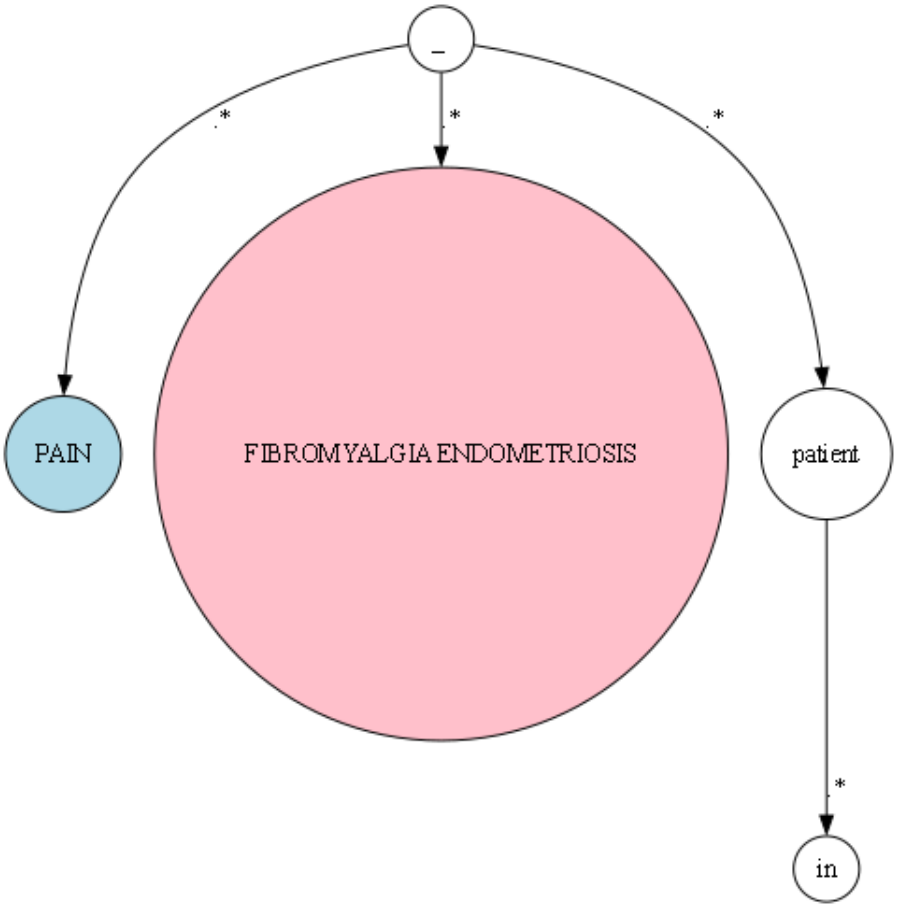



Figure III.19: New tagging - Pattern rule 24

```
(u_1 / .*
    :.* (u_13 / PAIN
        :entity (entity_0 / 1)
    )
    :.* (u_13 / FIBROMYALGIA|ENDOMETRIOSIS
        :entity (entity_1 / 2)
    )
    :.* (u_2 / patient
        :.* (u_3 / in)
    )
)
```

**Negative rule 1** Example sentence: #2066 - "YYY does not cause XXX although pain may limit muscular effort."

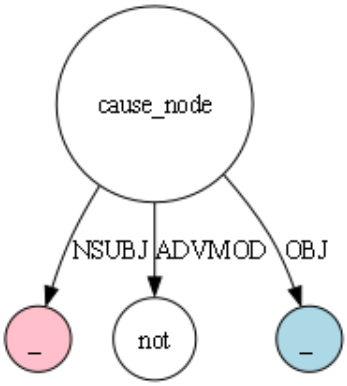



Figure III.20: New tagging - Negative rule 1

```
(u_3 / cause_node
    :NSUBJ (u_1 / .*
        :entity (entity_1 / 2)
    )
    :ADVMOD (u_2 / not)
    :OBJ (u_4 / .*
        :entity (entity_0 / 1)
    )
)
```

| | Train | | | Val | | |
|---|---|---|---|---|---|---|
| Rule | Prec | Recall | F1 | Prec | Recall | F1 |
| 1 | 0.625000 | 0.012428 | 0.024370 | 0.000000 | 0.000000 | 0.000000 |
| 2 | 0.716981 | 0.031483 | 0.060317 | 0.857143 | 0.040268 | 0.076923 |
| 3 | 0.900000 | 0.007457 | 0.014790 | 0.000000 | 0.000000 | 0.000000 |
| 4 | 1.000000 | 0.000829 | 0.001656 | 0.000000 | 0.000000 | 0.000000 |
| 5 | 1.000000 | 0.000829 | 0.001656 | 0.000000 | 0.000000 | 0.000000 |
| 6 | 1.000000 | 0.000829 | 0.001656 | 0.000000 | 0.000000 | 0.000000 |
| 7 | 1.000000 | 0.000829 | 0.001656 | 0.000000 | 0.000000 | 0.000000 |
| 8 | 1.000000 | 0.000829 | 0.001656 | 0.000000 | 0.000000 | 0.000000 |
| 9 | 0.750000 | 0.002486 | 0.004955 | 0.000000 | 0.000000 | 0.000000 |
| 10 | 1.000000 | 0.001657 | 0.003309 | 0.000000 | 0.000000 | 0.000000 |
| 11 | 0.540541 | 0.016570 | 0.032154 | 0.666667 | 0.026846 | 0.051612 |
| 12 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 |
| 13 | 1.000000 | 0.002486 | 0.004959 | 1.000000 | 0.006711 | 0.013333 |
| 14 | 1.000000 | 0.000829 | 0.001656 | 0.000000 | 0.000000 | 0.000000 |
| 15 | 1.000000 | 0.000829 | 0.001656 | 0.000000 | 0.000000 | 0.000000 |
| 16 | 0.782609 | 0.014913 | 0.029268 | 0.666667 | 0.026846 | 0.051613 |
| 17 | 1.000000 | 0.000829 | 0.001656 | 0.000000 | 0.000000 | 0.000000 |
| 18 | 1.000000 | 0.000829 | 0.001656 | 0.000000 | 0.000000 | 0.000000 |
| 19 | 0.777778 | 0.075394 | 0.137462 | 0.625000 | 0.033557 | 0.063694 |
| 20 | 0.672414 | 0.161558 | 0.260521 | 0.562500 | 0.120805 | 0.198895 |
| 21 | 0.597430 | 0.231152 | 0.333333 | 0.500000 | 0.181208 | 0.266010 |
| 22 | 1.000000 | 0.009114 | 0.018062 | 1.000000 | 0.006711 | 0.013333 |
| 23 | 0.894737 | 0.014085 | 0.027732 | 1.000000 | 0.013423 | 0.026490 |
| 24 | 1.000000 | 0.001657 | 0.003309 | 0.000000 | 0.000000 | 0.000000 |

Table III.1: Metrics after new tagging system

# Overview of Generative AI Tools Used

No generative AI tools have been used in writing this report.

# List of Figures

# List of Tables

# Bibliography


Banarescu, L., Bonial, C., Cai, S., Georgescu, M., Griffitt, K., Hermjakob, U., ... Schneider, N. (2013, August). Abstract Meaning Representation for sembanking. In A. Pareja-Lora, M. Liakata, & S. Dipper (Eds.), *Proceedings of the 7th linguistic annotation workshop and interoperability with discourse* (pp. 178–186). Sofia, Bulgaria: Association for Computational Linguistics. Retrieved from https://aclanthology.org/W13-2322

Bender, E. M., Gebru, T., McMillan-Major, A., & Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? . In *Proceedings of the 2021 acm conference on fairness, accountability, and transparency* (p. 610–623). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3442188.3445922 doi: 10.1145/3442188.3445922

Braganholo, V., Chirigati, F., Collberg, C., Culpepper, S., De Roure, D., Dittrich, J., ... Zobel, J. (2016, 01). Report from dagstuhl seminar 16041 reproducibility of data-oriented experiments in e-science. *Dagstuhl Reports*, *6*, 108-159.

Da, Y., Bossa, M. N., Berenguer, A. D., & Sahli, H. (2024). Reducing bias in sentiment analysis models through causal mediation analysis and targeted counterfactual training. *IEEE Access*, *12*, 10120-10134. doi: 10.1109/ACCESS.2024.3353056

de Marneffe, M.-C., Dozat, T., Silveira, N., Haverinen, K., Ginter, F., Nivre, J., & Manning, C. D. (2014, May). Universal Stanford dependencies: A cross-linguistic typology. In N. Calzolari et al. (Eds.), *Proceedings of the ninth international conference on language resources and evaluation (LREC'14)* (pp. 4585–4592). Reykjavik, Iceland: European Language Resources Association (ELRA). Retrieved from http://www.lrec-conf.org/proceedings/lrec2014/pdf/1062_Paper.pdf

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019, June). BERT: Pre-training of deep bidirectional transformers for language understanding. In J. Burstein, C. Doran, & T. Solorio (Eds.), *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)* (pp. 4171–4186). Minneapolis, Minnesota: Association for Computational Linguistics. Retrieved from https://aclanthology.org/N19-1423 doi: 10.18653/v1/N19-1423

Dumitrache, A., Aroyo, L., & Welty, C. (2018, jul). Crowdsourcing ground truth for medical relation extraction. *ACM Trans. Interact. Intell. Syst.*, *8*(2). Retrieved from https://doi.org/10.1145/3152889 doi: 10.1145/3152889

Fields, J., Chovanec, K., & Madiraju, P. (2024). A survey of text classification with transformers: How wide? how large? how long? how accurate? how expensive? how safe? *IEEE Access*, *12*, 6518-6531. doi: 10.1109/ACCESS.2024.3349952

Hassija, V., Chamola, V., Mahapatra, A., Singal, A., Goel, D., Huang, K., ... Hussain, A. (2023, 08). Interpreting black-box models: A review on explainable artificial intelligence. *Cognitive Computation*, *16*. doi: 10.1007/s12559-023-10179-8

Hendrickx, I., Kim, S. N., Kozareva, Z., Nakov, P., Ó Séaghdha, D., Padó, S., ... Szpakowicz, S. (2010, July). SemEval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In K. Erk & C. Strapparava (Eds.), *Proceedings of the 5th international workshop on semantic evaluation* (pp. 33–38). Uppsala, Sweden: Association for Computational Linguistics. Retrieved from https://aclanthology.org/S10-1006

Huang, L., Yu, W., Ma, W., Zhong, W., Feng, Z., Wang, H., ... Liu, T. (2023). *A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions.* Retrieved from https://arxiv.org/abs/2311.05232

Kornai, A., Ács, J., Makrai, M., Nemeskey, D. M., Pajkossy, K., & Recski, G. (2015, June). Competence in lexical semantics. In M. Palmer, G. Boleda, & P. Rosso (Eds.), *Proceedings of the fourth joint conference on lexical and computational semantics* (pp. 165–175). Denver, Colorado: Association for Computational Linguistics. Retrieved from https://aclanthology.org/S15-1019 doi: 10.18653/v1/S15-1019

Kovács, A., Gémes, K., Iklódi, E., & Recski, G. (2022). Potato: explainable information extraction framework. In *Proceedings of the 31st acm international conference on information & knowledge management* (p. 4897–4901). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3511808.3557196 doi: 10.1145/3511808.3557196

Lertvittayakumjorn, P., Choshen, L., Shnarch, E., & Toni, F. (2022, June). GrASP: A library for extracting and exploring human-interpretable textual patterns. In N. Calzolari et al. (Eds.), *Proceedings of the thirteenth language resources and evaluation conference* (pp. 6093–6103). Marseille, France: European Language Resources Association. Retrieved from https://aclanthology.org/2022.lrec-1.655

Levy, O., & Dagan, I. (2016, August). Annotating relation inference in context via question answering. In K. Erk & N. A. Smith (Eds.), *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 2: Short papers)* (pp. 249–255). Berlin, Germany: Association for Computational Linguistics. Retrieved from https://aclanthology.org/P16-2041 doi: 10.18653/v1/P16-2041

Manders, & Klaassen. (2019, 11). Unpacking the smart mobility concept in the dutch context based on a text mining approach. *Sustainability*, *11*, 6583. doi: 10.3390/su11236583

Matthiessen, C., & Bateman, J. (1991). *Text generation and systemic-functional linguistics: Experiences from english and japanese.* Pinter. Retrieved from https://books.google.at/books?id=f_RhAAAAMAAJ

Minaee, S., Mikolov, T., Nikzad, N., Chenaghlu, M., Socher, R., Amatriain, X., & Gao, J.

(2024). *Large language models: A survey.* Retrieved from https://arxiv.org/abs/2402.06196

Mozafari, M., Farahbakhsh, R., & Crespi, N. (2020, August). Hate speech detection and racial bias mitigation in social media based on bert model. *PLOS ONE*, *15*(8), e0237861. Retrieved from http://dx.doi.org/10.1371/journal.pone.0237861 doi: 10.1371/journal.pone.0237861

Pennington, J., Socher, R., & Manning, C. (2014, October). GloVe: Global vectors for word representation. In A. Moschitti, B. Pang, & W. Daelemans (Eds.), *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 1532–1543). Doha, Qatar: Association for Computational Linguistics. Retrieved from https://aclanthology.org/D14-1162 doi: 10.3115/v1/D14-1162

Qi, P., Zhang, Y., Zhang, Y., Bolton, J., & Manning, C. D. (2020). Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th annual meeting of the association for computational linguistics: System demonstrations.* Retrieved from https://nlp.stanford.edu/pubs/qi2020stanza.pdf

Qin, H., Li, M., Wang, J., & Wang, Q. (2024). *Adversarial robustness of open-source text classification models and fine-tuning chains.* Retrieved from https://arxiv.org/abs/2408.02963

Recski, G., Lellmann, B., Kovacs, A., & Hanbury, A. (2021). Explainable rule extraction via semantic graphs. In *Proceedings of the Fifth Workshop on Automated Semantic Analysis of Information in Legal Text (ASAIL 2021)* (pp. 24–35). São Paulo, Brazil: CEUR Workshop Proceedings. Retrieved from http://ceur-ws.org/Vol-2888/paper3.pdf

Ren, J., Xu, H., He, P., Cui, Y., Zeng, S., Zhang, J., ... Tang, J. (2024). *Copyright protection in generative ai: A technical perspective.* Retrieved from https://arxiv.org/abs/2402.02333

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). *"why should i trust you?": Explaining the predictions of any classifier.* Retrieved from https://arxiv.org/abs/1602.04938

Sen, P., Li, Y., Kandogan, E., Yang, Y., & Lasecki, W. (2019, July). HEIDL: Learning linguistic expressions with deep learning and human-in-the-loop. In M. R. Costa-jussà & E. Alfonseca (Eds.), *Proceedings of the 57th annual meeting of the association for computational linguistics: System demonstrations* (pp. 135–140). Florence, Italy: Association for Computational Linguistics. Retrieved from https://aclanthology.org/P19-3023 doi: 10.18653/v1/P19-3023

Shrikumar, A., Greenside, P., & Kundaje, A. (2019). *Learning important features through propagating activation differences.* Retrieved from https://arxiv.org/abs/1704.02685

Treude, C., & Hata, H. (2023). *She elicits requirements and he tests: Software engineering gender bias in large language models.* Retrieved from https://arxiv.org/abs/2303.10131

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I. (2023). *Attention is all you need.* Retrieved from https://arxiv.org/abs/1706.03762

Wang, J. T., Deng, Z., Chiba-Okabe, H., Barak, B., & Su, W. J. (2024). *An economic solution to copyright challenges of generative ai.* Retrieved from https://arxiv.org/abs/2404.13964

Wirth, R., & Hipp, J. (2000, 01). Crisp-dm: Towards a standard process model for data mining. *Proceedings of the 4th International Conference on the Practical Applications of Knowledge Discovery and Data Mining.*

Wu, B., Zhu, Z., Liu, L., Liu, Q., He, Z., & Lyu, S. (2024). *Attacks in adversarial machine learning: A systematic survey from the life-cycle perspective.* Retrieved from https://arxiv.org/abs/2302.09457

Ács Evelin, Ákos, H.-S., & Gábor, R. (2019). Parsing noun phrases with interpreted regular tree grammars. In *XV. Magyar Számítógépes Nyelvészeti Konferencia* (pp. 301–313). Retrieved from http://acta.bibl.u-szeged.hu/id/eprint/59094