Machine-Learning-Based and Rule-Based Clinical Information Extraction

Adrian C. Zevenster

DLMSAISCTAI01: Task 2 - AI in the Medical Domain

IUBH University of Applied Sciences

MSc(AI)

IT and Engineering

Contents

Abstract

Clinical Information Extraction has received growing interest, especially since the dissemination of Electronic Health Records. Relying on information extraction to discern meaning(structure) from written clinical documentation. Clinical text is nuanced, to deal with nuances, information extraction has relied on rule-based methods. This work looks at the natural language sub-tasks that underlay rule-based methods, underpinning it's contribution to machine-learning-based methods. A depiction to some of the NLP mechanisms to clinical information extraction is given, with an compendious on neural network architecture that has showed promise in clinical information extraction.

Keywords— Named Entity Recognition(NER), Relation Extraction(RE), Recurrent Neural Networks(RNN), rule-based, Machine Learning(ML)-based, Natural Language Processing(NLP), Clinical Named Entity Recognition(CNER), cTAKES, MedLEE, MetaMap, Deep Learning(DL), Long Short-Term Memory(LSTM), Clinical Information Extraction(IE)

1 Introduction

Clinical Information Extraction(IE) has seen tremendous throughput in the past decade. In-part due to adoption of electronic medical records(EMR), electronic health records(EHR), and other clinical documentation. Natural Language Processing(NLP) has been applied to a plethora of clinical IE task to assist domain experts. Clinical application ranges from patient phenotyping to adverse-drug event(ADE) identification.(Zeng, Deng, Li, Naumann, and Luo (2019)). These clinical IE solutions have predominantly been implemented as rule-based and expert-based systems. Machine learning(ML)-based systems has been investigated for the last decade yet remained contingent to clinical annotated corpora. The advent of Neural Networks(NN) and the age of Deep Learning(DL) has made the prospect of creating adaptable ML-based approaches more feasible, without the demand of regular feature engineering inherent to ML.(Hahn and Oleynik (2020)).

Although DL is often considered as an independent system; the interplay of various preliminary components are often overlooked - building upon ML-based and rule-based solutions. Many of the ML-based approaches rely on clinical corpora that has been meticulously developed through rule-based approaches and domain expertise. With time, the solutions have been refined, with ml methods assisting rule-based solution and visa-versa. DL describes the interaction of nodes present in NN architectures, however, in many cases information encoded in said nodes rely on ML algorithms and rule-based transformations to add value, especially in nuanced domains - such as clinical IE.

This paper aims to give an overview of data sources 2.1 involved in clinical IE, with an exploration and conceptualization of some of the most relevant clinical rule-based approaches 2.2 developed over the last decade. Most clinical information are presented in free-text format, that rely on computation linguistic to derive insights from text. NLP has been applied extensively to process text, in the case of clinical IE this branch of NLP relies on an amalgamation of transformations and rule-based approaches to extract meaningful from clinical information.

We aim to describe prominent algorithms associated to text transformation, especially transformations appropriate for ML, providing a short conceptualization 3, outlining the mechanics of these transformations

and their association to some rule-based, ML-based and DL-based approaches. Sets of transformations are typically what comprises the NLP process, with a variety of applications. This paper investigates components that are considered subtasks of NLP, concerning clinical IE two approaches are notable: Named Entity Recognition(NER) 4.2 and Relational Extraction(RE) 4.1.

Components associated to both NER and RE are outlined, illustrating how the aforementioned transformations, ML-based and rule-base approaches together construct the NER and RE pipeline encountered in clinical IE. Furthermore, we will describe some of the fundamental mechanics of DL-based methods - Recurrent Neural Networks(RNNs) 5.1 and Long short-term memory(LSTM) 5.2 - associated to clinical IE, which effectively assist in the creating adaptive and responsive NER and RE solutions for the ever evolving biomedical literature, with final discussion on ML-based and rule-based clinical IE, with suggestion on possible future improvements 6.

2 Clinical Information Extraction

Clinical IE has primarily employed rule-based methods to deal with extracting and identifying medical entities, with some consideration on ML-based methods.

Rule-based methods rely on knowledge-bases to govern rule sets within the rule-based system. These knowledge bases are usually domain experts that curate medical ontology's. From these ontology's rule-base methods are curated that identifies cases with similar logic expected from domain experts. Rule-based methods are constructed using components from NLP 4 incorporating various clinical corpora.

ML-based methods usually employs components from rule-based methods; with the possibility of improving past the limitations of rule-based methods; with the growing amounts of data generated, ML-based performance could potentially improve. ML reliance on copious amounts of data has remained a consideration, with rule-based methods outperforming ML-based solution. However since the emergence of Deep Learning(DL), ML-based solutions hold promise of matching performance achieved by rule-based methods, negating many drawbacks often associated to ML-based solutions.

Clinical IE application is mainly focused on processing of textual documentation; which includes clinical notes, laboratory reports, radiology reports, and pathology reports. Furthermore, EHR have assisted with the curation of patient summaries and patient profile extraction.

EHR has been used in conjunction with many systems to provide clinical insights; EHR in combination wit cTAKES 2.2.1 has been particularly useful for patient risk identification, along with other clinical tasks including ADE and genomic studies. EHR sources remained restricted the public - partly due to privacy concerns - with most EHR datasets kept internally; initiatives such as *i2b2*, *THYME*, and *MIMIC II* has created publicly available EHR on a variety of sources found from clinical documentation.

Primarily IE has been applied using knowledge-based and rule-bases methods, with comparatively little emphasis on ML-based methods. ML-Based methods has been applied to some clinical IE tasks and in some cases assisting already established rule-based methods. ML-based performance showed promise, however rule-based solution generalized to more clinically related tasks. DL 5 are one of advances that ignited interest in ML-based methods, specifically DL that is associated to NLP.

NLP has assisted clinical IE in a few sub-tasks: clinical syndromes, radiology reports, discharge summaries, problem lists, nursing documentation, and medical education documentation to list a few.(Wang et al. (2018)). IE contains several sub-tasks that can be considered, in clinical IE, this includes: Named Entity Recognition(NER), Relational Extraction(RE), Co-reference resolution, and Event Extraction. With promising results yielding from RE 4.2 and NER 4.2. There are clinical IE projects driven by NLP, many of these projects form the blueprint of further projects adopted on top of these established frameworks.

This section will elaborate on a few of the most widely adopted frameworks often implemented during the development of ML- and DL-base methods, including an overview of the data sources 2.1 often used during the curation of these systems. The systems/frameworks that are elaborated upon in section 2.2 emphasises clinical IE, in most cases relating to bio-informatics and biomedical information processing.

2.1 Data Sources

EHR and EMR has been meticulously implemented and maintained over the last few years. Along with an exceptional rise in clinical IE related research, which utilizes rule-base, expert-based and ML-based method. EHR is unique in that it contains free-text form data. Free text-from is more expressive in representation, often stored as unstructured text.

Clinical documentation are data sources that are often incorporated for IE related tasks, other sources that are often considered for IE include biomedical data and health related websites - here data mining is a primary tool, including PubMed or MEDLINE. Text mining is often used in conjunction with ML-based methods for IE; rule-based systems often make use of mining techniques, relying on text mining for the identification of certain rules. (Golshan, Dashti, Azizi, and Safari (2018)).

Clinical documentation is primarily divided into two categories: clinical notes, and diagnostic reports.

Clinical notes contains information about a patients(*data subject)* medical history, this includes social factors - which is correlated to patient health, physical examinations, treatment plans, diagnostics and summaries. IE could assist with the identification and elimination of variables in medical diagnoses that hinders the process of diagnosis.

Diagnostic is the second category related to clinical documentation, these diagnostic includes lab reports, radiology reports, and pathology reports. Which is a rich source of information in diagnostics, particularly if clinical notes and diagnostics reports are combined. Diagnostic documentation are often represented in form of brain imaging technologies(fMRI, X-rays, Ultrasounds) - IE methods are applied to both the signal analysis represented by these images - or summaries of reports created from these diagnostics can be extracted. (Wang et al. (2018)).

2.2 Clinical Information Extraction Systems

Tremendous collaboration to develop corpora and assistive tools tailored to clinical IE extraction has been undertaken, contemporaneously acting as a catalyst for the feasibility of ML-based and DL-based approaches for clinical IE.

Most of these collaborations were initially undertaken for the curation of rule-base methods, consisting

of annotated corpora, Metathesaurus and NLP frameworks. In many cases, these methods are conjoined with ML-methods assisting in creating more comprehensive models. This section outlines 3 of these collaboration efforts: cTAKES, MetaMap, and MedLEE.

These tools have assisted with the processing and identification of medical entities in various domains, using knowledge-base sources such as the Unified Medical Language System(UMLS) and various other medical related corpora. These are primarily rule-based undertaking, incorporating various NLP components.

Furthermore, these IE frameworks have assisted with creating richer clinical representation, in turn assisting with tasks such as creating word embeddings 3.3, succouring ML-based and DL-based methods - explored in later sections. These system have been prevalent in the construction of NER and RE models.

2.2.1 cTAKES

Built on Apache servers, clinical Text Analysis and Knowledge Extraction System(cTAKES), is an open-source NLP related system that has been developed for a plethora of clinical processing tasks, mainly used for text analysis and knowledge extraction which creates ontology's from the IE process, including phases of: co-reference resolution, NER, and RE to represent knowledge. (Bratanic (2021)).

cTAKES was build using the *Unstructured Information Management Architecture Framework(UIMA)*, another open-source framework combined with OpenNLP, creating structured data from unstructured sources. cTAKES contains engines for linguistics tasks and clinical tasks, which has been harnessed for various medical application: identification of phenotype cohorts, extraction of medical risk indicators - smoking, diabetes, exercise and diet history - detection of ADE, and identifying medication discrepancies constitutes some of the medical application. (Wang et al. (2018)).

cTAKES applies rule-based methods with NLP. cTAKES is amalgamated with ML-based methods, most frequently used as an NER model 4.2, which uses various computational methods to transform text, such as Tokenization 3.1, Part-of-Speech(POS) tagging 3.2, Normalization and Parsers. These are some of the annotators used to create cTAKES.

Using dictionaries that have already been created - such as the Unified Medical Language System(UMLS 2.2.2, and SNOWMED. cTAKES locates entities from text which are further related to concepts or terminologies, supplied from the created or embedded dictionaries.

2.2.2 MetaMap

MetaMap is an NLP tool which is primarily used to map text to the UMLS, often used for medical named entity recognition(CNER). MetaMap is a rule-based system, using vocabularies created through the UMLS.

The UMLS is a conspectus of biomedical terminologies, which is combined with NLP-methods and computational-linguistic techniques as medical text identifier.

The UMLS is a vocabulary database that contains biomedical and health related concepts; containing concept names and the relationships between their entities, the relation between separate entities are mapped, and recorded in a series of relational tables, with MetaMap applied to create a more granular representation. (NIH (2016))

The main semantic categories that the UMLS covers: disease, syndrome and clinical drugs related entities. The main parent class base for categorization include *Biological Function* which is divided into 2 primary leaves - Physiological, and Pathological Functions, all further sub-division are leave/children nodes to the primary leaves - such that all other categories are correlated to the primary categories.

The UMLS is primarily an indexing application which uses string matching, statistical processing, linguistic processing and POS tagging 3.2. UMLS includes vocabularies for Diagnosis, Procedure & Supplies, Disease and for comprehensive Vocabularies/Thesauri. (Humpherys, Leroy, Halper, and Bodenreider (2016))

MetaMap utilizes and builds on the UMLS framework an further maps biomedical text to the UMLS Metathesaurus, updating the conspectus entities. MetaMaps caters to multifarious clinical sub-genres, including: phenotype extraction, drug-disease treatment road-maps - most notably - along with assessment of EHR and extraction of patient related attributes. (Wang et al. (2018)) These tasks are primarily performed by the recognition and classification of medical texts such as clinical notes and summaries.

2.2.3 MedLEE

Medical Language Extraction and Encoding(MedLEE) is a clinical extraction framework with NLP-based methods which is used to extract, structure and encode clinical information.

MedLEE is used for a variety of clinical application, most notably; ADE detection, and finding trends and associations related to patient medical histories.

The encoder portion of MedLEE uses an combination of textual computation with knowledge-based components. Text and knowledge-bases are coalesced, ascribing the entities. During pre-processing text is segmented; this could be into sections, paragraphs or words, from which lexical attributes are determined.

The parser then uses the pre-processed text and creates structure with syntactic and semantic knowledge-bases. Within this phase, Parser Error Recovery is also included, in the event that the parser should fail, information recovery is attempted, preventing missing data - missing data of this kind is often difficult to identify by ML-based methods, leading to undesired model performance and potential introduction of bias in the model, the error recovery assists with these type of implications.

Phrase regularization is then used on the parsed text, this phase is used to ensure that non-contiguous word in text are put together, this is done by an composition table, which ensures that non-contiguous and contiguous words has the same output - entity description - during phrase regularization.

Expert-knowledge information is also added that is implicit to that specific domain. After regularization, encodings adds codes to the regularized outputs, this is done by mapping to an encoding table, such as the UMLS 2.2.2. (Shagina, Socratous, and Zeng (1996))

3 Extraction Techniques

There are several ways of approaching IE, particularly pertaining to medical orientated corpora, due the the domain specific nomenclature. Named Entity Recognition(NER 4.2) and Relational Extraction(RE 4.1) is often used for IE, especially in ML-based cases. NER and RE are both considered sub-tasks of

NLP 4. Knowledge Engineering is another technique that is often applied to IE related tasks. Knowledge Engineering amalgamates rule-based, expert-based, and data mining to produce rules for the extraction of meaningful information

This paper investigates IE techniques associated to ML based-methods, whilst touching on some of the transformation methods that are correspondent in both rule-based and ML-Methods, along with more recent IE developments in Deep Learning. Divisions of deep learning including: Natural Language Processing, Bioinformatics, Speech Recognition and Computer Vision. We will investigate some of these division and in particular delve into Recurrent Neural Networks 5.1; one of the neural networks that has been applied in NLP models, which makes it unique and critically important for information extraction task from a text based form.

Support Vector Machines(SVM) 3.4 and Conditional Random Fields(CRF 5.2) are ML-based methods that has performed reliably in clinical IE tasks; with comparitable performance to rule-based, and expert-based methods in extraction of medical related entities. SVMs achieved reliable performance for clinical IE tasks, with F1-Score 9.3 in the extraction of ADEs. Typically there are 4 categories that most drug-related events are prescribed to; Antibiotics, Anticoagulant, Insulin, and Opioid Analgesics. These ADE are the cause of approximately 1.3 million visits to the emergency room yearly. (CDC (2019)).

There are various methods used when extracting entities from unstructured data. These methods include: tokenization 3.1, POS tagging 3.2, TF-IDF vectorizers, n-grams, and Word2Vecto, naming but only a few.

To capture the message being portrayed a single IE method is rarely sufficient, instead a combination of the different extraction methods - each more apt for a specific task - is used and assembled to create a more accurate expression. These different methods are used in NLP models and in deep learning. They are also used in most of the rule-based and expert-based systems mention in section 2.

This section aims to describe some of the core components used for pre-processing text likely to be encountered in IE and some methods that are critical in handling nuances present in IE domains like clinical IE.

3.1 Tokenization

Tokenization typically splits sequences into chunks, in NLP, tokenization splits text, this process of splitting forms tokens. When splitting the word in free-text, the size of a token varies; various algorithms for text tokenization are available each different tokenizations rules. A new token can be indicated when punctuation is present - punctuation-based tokenizer - or when a blank space is present - Whitespace tokenizer - with many other variations.

Tokens can be divided into spans, this is essentially a sequence of tokens that when considered together reveal hidden relationships. Tokenization assists with identifying meaningful keys in text by considering the frequency of their occurrence, advantageous for both NER 4.2 and RE 4.1 models. Tokenization forms the core for many of the other pre-processing stages, such as the creating of word embeddings 3.3, which are essential in the creation of many rule-based systems such as MetaMap 2.2.2 and MedLEE 2.2.3.

Tokenization tranforms text data into acceptable numerical representations for NN inputs. During tokenization some text cleaning is also performed like the removal of stop words, which does not add any meaning to the interpretation of text, these word considered as 'filler'. Term Frequency Inverse Document Frequency(TF-IDF) is a common algorithm for implementing tokenization. Tokenization if often evaluated with cosine similarities or euclidean distance, where model performance model performance is evaluated with entropy 9.5 and purity. Tokenization is often considered as rudimentary but remains involved in IE solutions ranging from rule-based to DL-based methods.

3.2 Part of Speech Tagging

Part-of-Speech(POS) tagging marks entities and assign them appropriately, such as the identification of ER responses based on a response code. POS tagging is applied in both NER 4.2 and RE 4.1 for clinical IE, also involved in the systems described in section 2.2.

POS tagging makes certain assumptions from text based on the relevant semantics - these semantics are often supplied by medical vocabularies mentioned in section 2.2 - where POS is used as an entity identifier, which can be related to dictionaries, identifying relevant entities with their descriptions; these dictionaries are often supplemented by medical corpora. POS tagging assigns lexical terms to a word, which is then used to determine the syntax, this syntax can then be used for semantic processing, salient in both NER and RE.(Singh (2018))

POS tagging often serves as the syntax embedding inputs to DL models, the relevance of these inputs are described in section 5. POS tagging describes events from text: such as tens; whether a patient is still being treated, ailment status; cured or ill, and number; such as the amount of hospital visit made by the patient. POS taggers generally fall into two categories: rule-based and stochastic, of which Markov Chains are an example of the latter, often applied to improve POS tagging, more specifically, Hidden Markov Chains(HMM) are often used to improve performance. In clinical IE, POS tagging has been supported by initiatives to create their respective corpora, which are usually presented as pre-trained models, potentially decreasing the number of iterations required to achieve desired performance, there are also various methods employed for POS tagging; such as the perceptron, or transformation-based learner rules. (Pykes (2020)).

With POS tagging, the POS tag of the subsequent is determined based on the preceding POS tag; or the likelihood of the subsequent POS tag is based on the probabilities of the previous tags. This is usually determined by a transition matrix, illustrating utility of Markov Chains in POS related tasks - very similar to how RNNs5.1 operate to some degree. POS tagging is also prevalent in cTAKES, MedLEE and MetaMap; assisting with the identification and representation of clinical entities.

3.3 Word Embeddings

To be able to extract meaning from text computationally, text needs to be converted into a numerical representation, the caveat being; the semantics of the text needs to be preserved numerically. Algebraically, text is transformed into vector representations, referred to as vectorization. Simply defined, text is converted to a numerical representation. One-hot encoding takes the categorical variable, where each variable is text presented as an category, identified by tokenization 3.1 - or related methods, and is mapped

to an integer value, each integer values is then represented as a binary vector.

Vectorization methods like one-hot encoding are efficient, however, for large text datasets, applications of one-hot encoding has high dimensionality, another method often used for this vectorization process are word embeddings. Dimensionality requires solicitude when ML is involved(see section 9)

Word embeddings are based on sets of probabilities and co-occurances, where words are predicted based on what came before and/or after. (Pennington (2014)) A commonly used technique associate to NLP to create word embeddings are Word2Vec, which transforms text into vector representations, where the input is the text corpus and the output a set of vectors. With this array of numbers, matrix operation can be performed, to some respect this is the main function of creating word embeddings. Through the manipulation of these vectors, hidden relationships can be extracted. A vector contains more information that a single data point(scalar), when these vectors interact through matrix operations we create a higher dimensional representation without the added dimensionality. Embeddings are also used extensively in RNNs 5.1.

Word embeddings also give dense representations of tokens where preprocessing like one-hot encoding gives a sparse representation, denser representation conveys richer information than sparse representations. Another beneficial aspect of word embeddings: it allows for transfer learning, where the embeddings has a relevant weight matrix instead of an initial matrix, especially advantageous in clinical IE due to the specificity of the clinical corpus.

These embeddings can be used to share information across different clinical sub-genres, which could lead to less iterations and better performance, both for ML-based and rule-based approaches. (Kale et al. (2019)) GloVe, and Word2Vec are both pre-trained word embeddings that have been used to improve the UMLS and MetaMap 2.2.2 corpora.

Embedding models have also been used for NLP related IE; these models maps knowledge-bases into vector representations, these representations are then used to predict missing facts that could be present, this is done by extracting entities based on their extraction to latent variables, this is typically performed by matrix factorization - such as non-negative Matrix factorization are often used in recommender systems - and recent advancements in DL has used tensor decomposition to handle similar problems at higher dimensions. (Singh (2018)).

Embeddings are used as an input layer in DL and they are often employed in rule-based systems. They have shown to be greatly beneficial to RNNs, yielding better performance and more insightful models.

3.4 Support Vector Machines

Various components impedes the adoption of ML-based method, a frequent contributor is missing values; incorrectly captured data often leads to unintended bias in ML models, in the case of missing data in EHR, this causes inaccurate patient profiles and incorrect predictions, whether relating to ADE or phenotype profiling. Support Vector Machines(SVM) have been used extensively in clinical IE, both in rule-based and ML-based approaches. SVMs have been used for cancer classification models and have assisted both NER 4.2 and RE 4.1 related tasks.

When 2 classes need to be determined from a set of data point - one class might pertain to malignant and the other to benign cancer cells - Support Vector Classifiers(SVC) are used to obtain a boundary between the classes, referred as the *margin* separating the classes, acting as the threshold for the class identification criteria. The margin forms a boundary that separates the two classes(benign/malignant), when we allow the boundaries of these margins to overlap, we refer to the margin as a*soft margin*, this optimal overlapping region is what the SVC identifies.

The margin is formed by a series of intersecting hyperplanes, the modulus of the hyperplane determines the optimal distance between the classes. (Zhang, Xiao, and Gu (2019)). Improving the distance between the hyperplane is one of the main purpose of SVC, typically achieved with vector representations of data points that lie withing the identified hyperplane. Support Vector Machines(SVM) uses kernel functions to extrapolate the SVC functionalities into higher dimensions, without actually creating the higher dimensional data.

Polynomial kernels are applied to the vectors, transforming vectors into a higher dimensional representations. Dependant on the amount of classes, the polynomial function 2 is adapted, SVM optimizes the the distance - that is to say obtaining the optimal w (see appendices for hyperplane eq. 1) - between the hyperplanes, which are identified by the polynomial. Due to the nature of classification problems, the "bias-variance trade-off" becomes censorious.

The polynomial kernel computes the relationship of each observation in a higher dimension - where SVC is usually applied to lower dimensional data - thus more relationships are compared between each vector when applying the polynomial kernel. Cross Validation(CV) 9.4 is used to find the optimal degree(d) of the polynomial expressed in equation 2, where different degrees are iterated over to find the optimal performance. K-folds CV is usually applied to SVM optimization. SVMs are of particular use when it comes to: integrated feature-based classification - critial for clinical IE using ML-based methods - template based extraction, and automatic text classification for detecting adverse drug interactions .(Wang et al. (2018)).

SVM have been used extensively in cancer predictions tasks, using a Radial Basil Kernel Functions. Ghoulam, Barigou, and Belalem (2015) has combined SVMs with the MEDLINE and i2b2 corpora for semantic relations; where clinical tasks of disease treatment, and extraction of drug related adverse effects, achieving f1-scores of 91% and 87% respectively. (Caroll R. J. (2011)) has also suggested SVMs for the classification of Rheumatoid Arthritis, with the suggested model obtaining scores of 94% recall and 87% precision. SVMs has been applied and suggested in many clinical IE tasks relating to NER 4.2 and RE 4.1.

4 Natural Language Processing

NLP processes free-form text computationally. There are various components within the NLP pipeline, with IE considered a sub-task. Some of the methods mentioned within section 3 are considered subsystems of the constituting NLP pipeline, where subsystems are combined to form sub-tasks, and sub-tasks joined forming the NLP system.

NLP has been used for clinical language models and clinical IE. Medical extraction related topics that has been addressed in research and some application by sub-tasks of NLP include: Relation Extraction(RE),

Named Entity Recognition(NER) and Co-reference analysis. (Wang et al. (2018)).

RE typically uses some amalgamation of NLP components where NER refines separate components associated to NLP pipeline - components that are included within the NLP pipeline may include tokenization3.1, POS tagging3.2, and word embeddings3.3.

NLP has become one of the foremost methods used for structuring unstructured text, allowing for entity recognition, semantic extraction and identification. Apart from these applications, NLP has been on the forefront of IE, especially when considering clinical text processing. A primary concern is creating structured data, from here numerous other higher level system can be considered, such as machine translation, event extraction, and user(patient) profile extraction; covering various industry domains apart from clinical. This section consider two NLP components often encountered in the clincial IE pipeline: NER 4.2 and RE 4.1.

We aim to describe the functional contribution of both these components in relation to their prospective clinical praxis, outlining elements mentioned throughout this paper, chiefly in section 3, with their contributions to components present in NER an RE. Constituting component of NLP with relation to utilization and dependency on clinical corpora are presented, while considering NLP components in DL 5 and RNNs 5.1.

4.1 Relational Extraction

Relation extraction(RE) identifies pre-defined relationships between entities pairs, these could be entities described by rule-based or ML-based approaches. Fundamentally, RE is a method to transform unstructured text format into a structured format, while adding lexical and semantic fields.

RE typically employs components from POS tagging 3.2, dependency parsers and NER 4.2. Many clinical RE methods rely on supervised learning techniques, which requires large amounts of data. Thus large corporas' - or a multitude of corpora(2.2)- are required, especially for ML-based RE, fortunately, many public health initiatives has aided in meeting this requirement, as mention in section 2.

However, data volumes required for model training still pose many challenges. As remedy for this impediment, at-least in the case of RE, Distant Supervision(DS) has been suggested. DS augments the amount of training available by adding additional resources to the knowledge-base, creating tags from available resources, such as PubMed 2. The UMLS 2.2.2 database already uses PubMed as a source, along with many others. (Singh (2018)).

DL methods have also been implemented improve RE related task, including RNNs 5.1, Convolutional Neural Networks(CNN), and Long Short-Term Memory(LSTM) networks 5.2 - a gated network architecture used in conjunction with the neural network - has been used to improve and facilitate RE tasks. Word embeddings 3.3 - such as Word2Vec - have also been used in RE tasks, with similar performance to some DL methods, which has the benefit of needing less computational resources and less data when initially training the model.

RE relies heavily on engineering features and kernel methods; such as those mention in relation to SVMs 3.4. The process of RE usually involves information extraction from the relevant sources, sources

and extractions are then classified, based on some confidence calculation, classifications are then ranked based on the criteria of the confidence interval. (Golshan et al. (2018)).

When relational entities in the clinical domain are considered there are typically two categories: paradigmatic relations and syntagmatic relations.

Paradigmatic entities are entities of the same task, this usually include antonyms or synonyms of a given entity, where syntagmatic relations are links between separate entities, this could include relation between treatment entities from extracted disease entities. SVMs have been shown to assist in relatable tasks.

(Magge, Scotch, and Gonzalez-Hernandez (2018) has suggested a combination of LSTM 5.2 an Random Forests - an ensemble learning method - to identify entity relations. This was done by filtering entity pairs; then features were extracted from identified entities using Random Forests, achieving an F1-micro-average score of 88% for clinical task such as dosage extraction, adverse medication interactions, treatment duration, treatment reason, and manner of treatment. RE has also shown to be tightly linked to NER 4.2, with many of the state-of-the art NER models building from the entities extracted and identified during the RE process.

4.2 Named Entity Recognition

Name Entity Recognition(NER); a primordial step in the NLP pipeline, is considered as a word-level tagging, wherein each word in a sentence is mapped to a named entity tag. NER has been applied vastly in clinical IE, during the development of rule-based, ML-based and DL-based methods.

NER models are generally comprised of: Tokenization 3.1, Gazetteers, Word Embeddings 3.3, and POS tagging 3.2. NER is a considered a supervised learning task, including: Hidden Markov Chains (3.2, SVMs 3.4, Maximum Entropy Models, Decision Trees, and Conditional Random Fieldd(CRF) (5.2, 9.5), other ML techniques including unsupervised and reinforcement learning are often utilized.

NER models assist with the identification of different medical entities pertaining to a certain corpus. The entities that are extracted include entities relating to: treatment, test, disease, symptom, medication, and drug name. (Ghoulam et al. (2015)).

ML-based NER models are often suggested, although it has been show that rule-based NER models has somewhat better performance 6, ML-based approaches has the advantage of 'learning' new entities from training data, moreover, ML assisted rule-based methods are common place.

Yang, Liu, Qian, Guan, and Yuan (2019) found that clinical NER has traditionally been rule-based, incorporating MedLEE 2.2.3, cTAKES 2.2.1, and KnowledgeMap. RE 4.1 is fundamentally built upon the NER model, where RE provides more context to the entity classification provided from the NER model.

Yang et al. (2019) has suggested a faceted approach for creating NER models, coalescing rule-based systems, feature engineering, and DL; rule-based systems relies on search patterns to provided context to entities, ML-based system can then be used to learn patterns, feature engineering becomes vital for any ML-based solutions from here DL can be applied to improve on existing solution.

SVMs and word embeddings have been extensively applied to assist NER performance, DL has also shown

promising performance for NER models. DL architecures that are frequently used includes RNNs 5.1, Bi-LSTMs 5.2, and CRF.

(Hahn and Oleynik (2020)) has suggested that DL-based NER outperforms all other methods for identification of disease mentions, although not the most comprehensive, performance improvements are notable. Gorinski et al. (2019) has compared rules-based(EdIE-R) and ML-based(EdIE-N) NER solutions for radiology reports, where EdIE-N was able to infer named entity annotations from the training data provided, however EdIE-R maintained a slight edge in performance. EdIE-N was assisted by DL, incorporating both CRF and LSTM 5.2.

Furthermore, EdIE-N incorporated no external sources (see section 2.2). Furthermore, results has shown, as clinical data stores increase, performance on ML-based solutions can be tweaked; however the introduction of unnecessary dimensionality is relevant when considering this approach. NER is influential in most of the clinical IE approaches that have been considered - as the clinical corpora continues to expand so does the entities with their related semantic maps(descriptions of entities), emphasizing to advantage of entities that can be added and updated as the corpora expands automatically.

5 Deep Learning

The term Deep Learning(DL) and Neural Networks(NN) are used somewhat interchangeably. Artificial Neural Network(ANN) is a basic NN architecture; comprised of an input layer, hidden layers and output layers - each layer is further comprised of nodes, the NN architecture act as the schematic of the interactions and the transformations performed the layers and nodes within layers. DL becomes important once the ANN gains more complexities and more layers are present. Different NN architecture are considered in relation to the task, such as machine translation or image processing.

RNN and CNN are considered DL architectures, which describes the layers and their interactions within the architecture. NNs typically pertains to the functionalities of the NN at hand; this includes aspects such as the learning rate, activation functions, and weights; where DL describe how to structure layers within the given architecture. One of the DL approaches often used for clinical text processing are RNN, tailored for handling sequential data, extensively employed in clinical IE, even improving upon already presented clinical IE solutions, often rule-based.(Hahn and Oleynik (2020))

In this section we will investigate the preferred NNs for NER 4.2 and RE 4.1 tasks when it pertains to clinical text processing. Emphasis is put on the NN architecture that have been auspicious for NER and RE tasks, namely RNNs 5.1 and LSTM 5.2 architectures. With transient consideration how DL approaches could be considered in improving existing models, such as CNN (see section 6).

5.1 Recurrent Neural Networks

Recurrent Neural Networks(RNN) has been instrumental in text processing - a source of sequential data. (Chollet (2017)).

RNNs are customarily used for sequential data, however an unique aspect are inherent to RNN, in which the hidden states of the network stores information of the sequence.(Golshan et al. (2018)).

Early NNs are based on principals of Feed Forward Networks(FFN), where information only flows in a forward direction. In FFN the information from the input node flows through hidden layers to the output layer, thus the network is not concerned with the state of any previous nodes. Whereas RNN considers the state of the previous node. The process of considering the previous state of a node is generally referred to as *back-propagation*.

The are various variation of RNNs architectures, based on the required task, such as: many-to-one, one-to-many or many-to-many. Different ratios are beneficial to different model requirements. In many-to-many sequences, the current node considers previous nodes input and output, thus different sequences are more suitable dependant on the task.(Geron (2019)).

RNNs are typically comprised of: input layers, hidden layers/states, activation functions, weights and outputs. $x(t)$ is taken as the input at every timestep(t indicates the timestep), such that $x(t-1)$, where some portion of the sequence is considered. The hidden states acts as the 'memory' function; storing the information from the previous state, used as an input in the preceding node.(The hidden state is presented formally in eq 3).

The output applies some activation function to generate an interpretable output of probabilities associated to each predicted outcome. Commonly, RNN can be considered as an amalgamation of forward- and backward-propagation. During forward propagation the state of each time-step is considered; including the input, weight associated and activation function, from here a loss function, such as cross-entropy 9.5, is used to determine the 'quality' of the prediction in each node, achieved by the sum of the loss function over the entire model sequence. (David Cecchini (2021)).

Cross-entropy is often applied to classification problems as it measures the difference between the predicted and actual outcomes, even when given as a probability distribution. Once the loss functions has been computed, backward-propagation calculates the derivatives of the loss functions for each time-steps to respects to the weight matrices(see appendix for derivative description 5, and matrix descriptions 7). This process often leads to *Exploding and Vanishing Gradients*, where the derivative converges to 0 or diverges to ∞ - neither gives any actionable results.

To mitigate this phenomena, various methods have been suggested such as gradient clipping in the case of exploding gradients; with ReLU, GRU and LSTM cells, favored for handling vanishing gradients. Activation functions are another important consideration for NN architectures, activation functions are responsible for transforming the summed weights from one node into an input of the following node, or acting as the the output of a node.

LSTM have been particularly beneficial for clinical IE related tasks; especially pertaining to NER4.2 and RE4.1. Apart from handling vanishing gradients, LSTM can consider more information in a longer sequence; detecting relational utterances to an entity, and the classification of entity relation into defined classes are typical clinical RE tasks that require long-term dependencies.

In principal, the memory(hidden state) is represented as $h(t)$ (see eq. 3). U is the input matrix to the NN; this is usually vectorized data such as word embedding 3.3, that serves as the input to the first layer, W and V are weight matrices present in the hidden layers and output layer (see appendices 7 for formal RNN description). Back-propagation is imperative to updating the associated weight matrices, allowing

for the updating of the cell memory. Nabi (2019).

Singh (2018) has shown that RNN are used in state-of-the-art RE models use in clinical IE, and RNNs (specifically LSTM 5.2) are applied to state-of-the-art CNER models.

5.2 Long Short-Term Memory

Long Short-Term Memory(LSTM) address limitation presented in 'vanilla' RNN architectures. In clinical IE the LSTM model combined with Conditional Random Fields(LSTM-CRF) has shown significant improvements on NER 4.2 based tasks. The LSTM gate acts as feature extraction and CRF adds an additional layer which infers sequence labels from clinical documentation 2.1. (Yang et al. (2019))

LSTM cells consists of various gates that, taking inputs from the previous cell state. LSTM are used to address the exploding/vanishing gradients 5.1, prevalent in RNNs - distinctly the vanishing gradient. Typically the LSTM cell contains 3 gates: forget Gate, input Gate, and output Gate.

The forget gate acts as a sentry for the previous cell state, the forget gate identifies exponential increase or decrease of the of the gradient from the loss function, the forget gate acts as the remediator. The output gate combined with some activation function, typically a non-linear activation function, this output is used as the output to the next cell state.

This is particularly advantageous for identifying many clinically intricate extractions; such as drug interaction, adverse drug interactions. LSTM architecture have been suggested to self-diagnosing assistants, where LSTM can consider various interactions dependant on some specified parameters. (Loy (2019)).

Condition Random Fields(CRF), a model that has been beneficial to many NLP related tasks, where the 'conditional' is a conditional probability distribution, which can be considered as an extension of Hidden Markov Models(HMM) 3.2.

CRF adds a condition boundary around the probability states presented from the HMM model. CRF has been combined with LSTM to NER 4.2 and RE 4.1 tasks, clinically showing potential for a plethora of tasks ranging form ADE to dosage identification.

CRF and RNN amalgamated models have been suggested by Magge et al. (2018) for these kind of undertaking, achieving a micro-average F-score of 81% for NER and 88% for RE. (See appendices for related medical entities 8). LSTM has been applied to the processing of clinical documentation 2.1, extraction of patient details from EHR. According to Habibi et al, biLSTM-CRF is considered as state-of-the-art in clinical NER 4.2.

6 Discussion

Clinical IE has shown great promise with the advent of Deep Learning. With the performance of Alpha Fold at the Critical Assessment of Protein Structure Prediction(CASP) event in 2020, outperforming all previous attempt at the problem of 3-D protein folding. Establishing that the revolution of DL is could truly be on the horizon; and in many cases re-establishing the primary role that ML could play. (Google (2019)).

NLP relies heavily on knowledge sources - along with rule-based sources - when creating RE 4.1 and NER 4.2 models, whereas DL relies less on these sources, however with the drawback of needing substantially more training data (Wu et al. (2019))

Gated recurrent Units(GRU) have been shown to deal with vanishing/exploding gradients 5.2 achieving similar performance to LSTM cells in dealing with vanishing gradients. However on clinical IE extraction, performance is not as clear.

Bias and noisy data has been shown to be detrimental to precision, something that EHR 2.1 is prone to, discussion on appropriately handling such nuances could greatly assist ML-based and DL-based models. Purely ML-based methods - as mentioned in section 3 - has incredulous parameters and features that need to continuous be adjusted and re-evaluated; which in turn could add granularity in creation of the model- which could give richer information on patient profiles or more description on classifications such as when identifying benign or malignant cancer, however, most feature engineering efforts have unsalable improvements on performance in comparison to the effort required, whereas DL-based methods 5 require far less attention in feature engineering for similar and in many cases improved performance; with the downside of needing vastly more amount of data to consider a feasible model. The learning curve for DL-based methods in higher compared to ML-based, needing far more volumes of data and more training iterations.

Gazetteer features has been suggested to improve both NER and RE related tasks.(Magge et al. (2018)).

Gazetteer is a list of pre-defined values that all relate to the same entity. Gazetteers' can be constructed from medical domain-knowledge corpora mention in section 2, however the growing clinical corpus can make this a time consuming endeavour, thus considering a model that continuous learns, that is to say, as the corpora expands so does the parameters of the model could be considered to further improve potential models.

In general, rule-based solutions still outperform ML- and DL-based methods. Nonetheless, DL is incrementally closing the gap in performance between ML-based and rule-based solutions. Furthermore, there are continuous developments to further potentially remedy ailments associated to ML learning methods, such as Distant Supervision 4.2, in the case of supervised learning.

Transfer learning, a recent topic in clinical IE has been shown to assist both NER and RE 4.1 models - considered as a branch of semi-supervised learning. Transfer learning allows for vocabularies - such as those mentioned in section 2.2 - to be compounded with new entities and associated semantics over iterations, transfer learning relies heavily on word embeddings 3.3. Singh (2018).

Although rule-based systems achieves gold standard performance in clinical IE, the various improvements and expansion presented by ML and DL are indeed optimistic. ML-based methods could see substantial increases in performance as available data increases, in some cases ML-based solutions may even be pertinant in handling amount of data that needs to be structured

References

Bratanic, T. (2021, 02). *From text to knowledge. the information extraction pipeline.* Retrieved from https://towardsdatascience.com/from-text-to-knowledge-the-information-extraction-pipeline-b65e7e30273e

Caroll R. J., D. J. C., Eyler A. E. (2011). Naïve electronic health record phenotype identification for rheumatoid arthritis. *AMIA -Annual Symposium Proceedings. AMIA Symposium*, 189-96.

CDC. (2019). *Adverse drug events from specific medicines.* Retrieved from https://www.cdc.gov/medicationsafety/adverse-drug-events-specific-medicines.html

Chollet, F. (2017). *Deep learning with python* (1st ed.). USA: Manning Publications Co.

Cunningham, P., & Delany, S. J. (2020). Algorithmic bias and regularisation in machine learning. Retrieved from http://search.ebscohost.com.pxz.iubh.de:8080/login.aspx?direct=true&db=edsarx&AN=edsarx.2005.09052&site=eds-live&scope=site

David Cecchini, A. S., Chester Ismay. (2021). Recurrent neural networks for language modelling with python.. Retrieved from https://app.datacamp.com/learn/courses/recurrent-neural-networks-for-language-modeling-in-python

Geron, A. (2019). *Hands-on machine learning with scikit-learn and tensorflow concepts, tools, and techniques to build intelligent systems.* O'Reilly Media, Inc.

Ghoulam, A., Barigou, F., & Belalem, G. (2015, 04). Information extraction in the medical domain. *Journal of Information Technology Research*, *8*, 1-15. doi: 10.4018/jitr.2015040101

Golshan, P. N., Dashti, H. R., Azizi, S., & Safari, L. (2018). A study of recent contributions on information extraction. *CoRR*, *abs/1803.05667*. Retrieved from http://arxiv.org/abs/1803.05667

Google, D. (2019). *Alphafold protein structure database.* Retrieved from https://alphafold.ebi.ac.uk/

Gorinski, P. J., Wu, H., Grover, C., Tobin, R., Talbot, C., Whalley, H., ... Alex, B. (2019). Named entity recognition for electronic health records: A comparison of rule-based and machine learning approaches. *CoRR*, *abs/1903.03985*. Retrieved from http://arxiv.org/abs/1903.03985

Hahn, U., & Oleynik, M. (2020, 08). Medical information extraction in the age of deep learning. *Yearbook of Medical Informatics*, *29*, 208-220. doi: 10.1055/s-0040-1702001

Humpherys, B., Leroy, G., Halper, M., & Bodenreider, O. (2016). *Source vocabularies.*

Jincymol Joseph, J. R. J. (2018, 11). Information extraction using tokenization and clustering methods. *International Journal of Recent Technology and Engineering*, *8*, 3.

Kale, M., Siddhant, A., Nag, S., Parik, R., Grabmair, M., & Tomasic, A. (2019). *Supervised contextual embeddings for transfer learning in natural language processing tasks.*

Kerzel, P. U. (2020). *Use case and evaluation.* IUBH Internationale Hochschule GmbH.

Loy, J. (2019). *Neural network projects with python : The ultimate guide to using python to explore the true power of neural networks through six projects.* Packt Publishing.

Magge, A., Scotch, M., & Gonzalez-Hernandez, G. (2018, 04 May). Clinical ner and relation extraction using bi-char-lstms and random forest classifiers. In F. Liu, A. Jagannatha, & H. Yu (Eds.), *Proceedings of the 1st international workshop on medication and adverse drug event detection* (Vol. 90, pp. 25–30). PMLR. Retrieved from https://proceedings.mlr.press/v90/magge18a.html

Maté, G. (2011). *Scattered minds: The origins and healing of attention deficit disorder.* Knopf Canada.

Retrieved from https://books.google.co.za/books?id=_FoNmxfSPAUC

Nabi, J. (2019, 07). *Recurrent neural networks (rnns).* Retrieved from https://towardsdatascience.com/recurrent-neural-networks-rnns-3f06d7653a85

NIH. (2016). *Unified medical language system (umls).* Retrieved from https://www.nlm.nih.gov/research/umls/index.html

Olawale F., A., Lisa, Z., Tolulope T., S., Richard, S., Eric, B., & Lisa M., L. (2019). Impact of missing data on bias and precision when estimating change in patient-reported outcomes from a clinical registry. Retrieved from http://search.ebscohost.com.pxz.iubh.de:8080/login.aspx?direct=true&db=edsbas&AN=edsbas.35C59D67&site=eds-live&scope=site

Pennington, J. (2014). *Glove: Global vectors for word representation.* Retrieved from https://nlp.stanford.edu/projects/glove/

Pykes, K. (2020, 11). *Part of speech tagging for beginners.* Retrieved from https://towardsdatascience.com/part-of-speech-tagging-for-beginners-3a0754b2ebba

Shagina, L., Socratous, S., & Zeng, X. (1996, 01). A web-based version of medlee: A medical language extraction and encoding system. *Proceedings of the AMIA Fall Symposium*.

Shaikh, R. (2018, 11). *Cross validation explained: Evaluating estimator performance.* Towards Data Science. Retrieved from https://towardsdatascience.com/cross-validation-explained-evaluating-estimator-performance-e51e5430ff85

Singh, S. (2018). Natural language processing for information extraction. *CoRR*, *abs/1807.02383*. Retrieved from http://arxiv.org/abs/1807.02383

Wang, Y., Wang, L., Rastegar-Mojarad, M., Moon, S., Shen, F., Afzal, N., ... Liu, H. (2018). Clinical information extraction applications: A literature review. *Journal of Biomedical Informatics*, *77*, 34-49. Retrieved from https://www.sciencedirect.com/science/article/pii/S1532046417302563 doi: https://doi.org/10.1016/j.jbi.2017.11.011

Wu, S., Roberts, K., Datta, S., Du, J., Ji, Z., Si, Y., ... Xu, H. (2019, 12). Deep learning in clinical natural language processing: a methodical review. *Journal of the American Medical Informatics Association*, *27*, 457-470. Retrieved 2021-07-12, from https://academic.oup.com/jamia/article-abstract/27/3/457/5651084 doi: 10.1093/jamia/ocz200

Yang, J., Liu, Y., Qian, M., Guan, C., & Yuan, X. (2019, 09). Information extraction from electronic medical records using multitask recurrent neural network with contextual word embedding. *Applied Sciences*, *9*, 3658. doi: 10.3390/app9183658

Zeng, Z., Deng, Y., Li, X., Naumann, T., & Luo, Y. (2019, 01). Natural language processing for ehr-based computational phenotyping. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, *16*, 139-153. doi: 10.1109/tcbb.2018.2849968

Zhang, X., Xiao, J., & Gu, F. (2019). Applying support vector machine to electronic health records for cancer classification. In *Proceedings of the modeling and simulation in medicine symposium.* San Diego, CA, USA: Society for Computer Simulation International.

A Data Sources

As with most knowledge domains, understanding and interpreting nuances are pertinent. Keeping track of patient medical history, and the latest medical research to most effectively treat a patient is arduous. In most cases these nuances are documented, however relating observation do disparate entities become muddled.

To illustrate this concept in relation to ADE identifications - a clinical task frequently addressed in clinical IE, which NLP 4 and is various sub-tasks address - consider the case of a patient with a treatment plan of Concerta and/or any form of SSRIs; from the prescription of Conerta, it could be inferred that the patient suffers from ADD/ADHD, which correlates to irregular levels of dopamine, serotonin, or oxytocin - chemicals in the brain that regulated brain function - such patients are generally more susceptible to depression or related ailments, hence the treatment with SSRIs. However, absent reports of depressive symptoms at the initial diagnosis with symptoms reported after a time-period of using ADD/ADHD medication could be indicative of side affect associated with the amphetamine medication, or a genetic component might be relevant in the patient.(Maté (2011))

Assisting with these type of task leaves room for clinical experts to focus on treating the patient most effectively.

B Support Vector Machines

SVMs are supervised learning tasks, a learning algorithm used for classification tasks. SVC - a precursor to the SVM - are used to determine the optimal hyperplanes between classes. The SVC forms a boundary around each class, creating a classification criteria boundary. The SVC finds the maximal distance between two classes before the boundary threshold changes, this any data point that falls withing the specified criteria will belong to that class. The formal definition of a general hyperplane is presented in eq 1.

$$w \cdot x + b = 0 \tag{1}$$

Equation 1 represent the hyperplane between two classes, where w indicates the vector normal to the hyperplane, and b indicated the offset. The hyperplane is the boundary that separates the two classes, based on some vectors present in the provided data - described as support vectors. The hyperplane forms the decision boundary 3.4 of SVM and is created from the input data, hyperplane(s) are extended into higher dimension when the kernel function is applied. Thus finding the hyperplanes that maximizes the distance between classes are essential before the kernel function is considered. Zhang et al. (2019)

The kernel function is applied to the SVC mention in section 3.4, the function is used to represent data in a higher dimensional feature space, the kernel function is used as a stereoscope; transforming vector representations - such as clinical entities- into higher dimension representation with the aim of gaining additional insight into the perspective entities(clinical entities) , at this stage we refer to the SVC as SVM. There are various kernel functions, the most common is usually the polynomial kernel 2, usually used when we are considering 2 classes.

$$K(x, y) = (a \times b + r)^d \tag{2}$$

$$Where: \quad a, b = observations$$

$$r = coefficient$$

$$d = degree$$

$k(x, y)$ represents the kernel function applied to both x and y, which could be a representation of data points, vectors or even functions. Where r indicates the coefficient of the polynomial, specifying the parameters, and d is the degree to which the polynomial is raised.

7 Recurrent Neural Networks

RNN has been applied to a variety of information extraction tasks, namely named entity recognition and relation extraction, which clinically has been applied to task such as adverse-drug identification, ...

RNNs are comprmised of: input layers, output layers and hidden layers.

$$h(t) = f(Ux(t) + Wh(t-1)) \tag{3}$$

The hidden state of the network is presented in eq. 3: U, V, and W represent matrices, where U is act as the input matrix, W the hidden state weight matrix. Which is then applied to a activation function such as $tanh$, which are imperative for the 'learning' nature associated to RNNs. In the case of RNN, a non-linear activation function is preferred.

$$\begin{aligned} o^t &= c + V(h)^t \\ \hat{y}^t &= softmax(o^t) \end{aligned} \tag{4}$$

Equation 4 shows the output function associated to weight matrix V 5.1, from which another activation function is applied to achieve an intepretable output presented as $\hat{y}^t$, where the activation function is applied to the output function. Softmax is often used as it coverts vectors into probabilities.

$$\frac{\partial a_t}{\partial W_a} = (W_a)^{t-1} g(X) \tag{5}$$

Back-propagation equation, where derivative over the layers are summed as an error, this allows for weight associated to layer in the NN architecture to be updated and optimized, allowing for improvement of performance with each iteration.

8 Long Short-Term Memory

Model results using Bi-Char-LSTMs-CRF presentened by Magge et al. (2018), where Conditional Random Fields are combined with Long Short-Term Memory, specifically Bi-Character LSTM gates, where every second character is considered in model predictions. This figure represent the Recall, Precision and F1-scores achieved for NLP sub tasks: Named Entity Recognition and Relational Extraction based on a presented clinical task. Figure represents the sub-task associated subset of clinical task.

This model achieved a Micro-Average F1-score of 81% for NER of medical entities, with a Micro-Avg F1-score of 88% for RE; medical entities with their associated sub-task are presented in table 1

Task	Label
NER	Drug
	Dose
	Route
	Frequency
	Duration
	Indication
	Severity
	SSLIF
	ADE
RE	Dosage
	Manner
	Route
	Frequency
	Severity
	Type
	Reasoon
	Adverse

Table 1: Medical entities identified by sub-task from th Bi-Char-LSTM-CRF model

9 Evaluation

Classification metrics are typically based on 4 categories: True Positive(TP) - the relevant positive item is correctly classified, True Negative(TN) - the relevant negative item is incorrectly classified, False Positive(FP) - when the relevant negative item is correctly classified, False Negative(FN) - when relevant negative item is incorrectly classified. The summary of these classifications are usually presented in a *Confusion Matrix*, providing a holistic representation of classification performance, from which precision and recall can be calculated; other measures can also be calculated such as sensitivity and specificity.

Models that require continuous evaluation require metrics such as mean squared error, mean absolute deviation or root mean squared error to measure performance, classification performance metrics usually include: precision, recall and F1-scores, these measures are elaborated upon within this section. These metrics contain 4 categories: True Positive(TP), True Negative(TN), False Positive(FP), and False Negative(FN).(see appendix 9 for further elaboration). These classification metrics has been implemented as performance measures in fundamentally all clinical IE related tasks; whether the methods pertain to rule-based, ML-based or DL-based. Furthermore, these metrics are used to evaluated many NLP models; including NER 4.2 and RE 4.1 sub-tasks, many NN use these the same metrics to evaluate performance; especially RNNs 5.1 and LSTM 5.2. From these classification categories, inferences can readily be made of where predictions and classifications are cumbersome.From a ML-narrative, these metrics make it possible to derive possible overfitting or underfitting present in models, assisting with identifying potential bias within the model. (Cunningham and Delany (2020))

The *bias-variance trade-off* becomes imperative when ML models are considered, where two states are considered: bias and variance. Bias essentially is the processes of assigning weights to features; high bias indicates that a few set of features has higher weight associated to them than other features. Whereas variance measures the area of effect, in models with high variance, the model predictions disproportionately favours the training data, the model closely mimics the data on which it has been trained, failing to generalize on unseen cases - often indicated by poor performance on testing data.(Olawale F. et al. (2019)) Rule-based and expert-based systems generally have high precision scores 7 with relatively low recall scores 6 for clinical NER and RE. ML-based methods have shown markedly better recall performance, with marginally lower recall when considering clinical NER and RE.(Yang et al. (2019))
Precision and recall are used not only in determining classification performance but utilized when measuring ML-based classification tasks like SVMs3.4, decision trees, and random forests. Precision is used to measure NER performance. This section aims to give a brief overview of these measures, to aid the interpretability of mentioned metrics. Furthermore, the measures discussed give some initiative grasp on the utility on models that include Neural Networks 5.1 and DL. Further, we aim do correspond these evaluation metrics to application mention for clinical RE and NER.

9.1 Recall

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

Recall; the measure of TP classification over the sum of TP and FN classifications. Recall is useful for the determination of the amount of 'meaningful' classifications. Recall is used as an evaluation proxy in classification tasks, and for the purposes of this paper it is very often applied to NLP model performance metrics.

Recall is the measure of True Positive(TP) 9 classifications over the sum of TP and False Positive(FP) 9 classifications 6. Whereas precision measures TP classifications over TP and FP classifications 7. Precision gives a measure on how many of the positive classifications are correctly identified. Considering the case of drug-related emergency room visits; not all ER visits will be indicative of drug-related problem, precision gives a measure of how confident we are that the ER visit described as drug-related ended up being exact.

9.2 Precision

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

Precision; the measure of true positive classifications over the sum of true positive and false positive classification. Precision can be described as the accuracy of positive predictions. These can be particularly important when considering , for instance, cancer classifications, where positive classifications needs to be accurate with little room for error.

Precision can be considered as a measure of how useful the results of the classification are, or how complete the results are in the holistic representation. Precision measures the true positives over the sum of true positives and false positives, shown in eq. 7. (Kerzel (2020)) Precision describes the accuracy of positive predictions. Precision is an indication on how much error is caused by false negative(FN) classifications.

Dependant on the nature of the use-case precision or recall may be desired; a paramount decision due to the somewhat inversely correlated nature present in these suggested metrics.

9.3 F1-Score

Dependant on the use case, the rate of FP or FN might be more/less than undesirable than the other. The model optimizes to ensure there are fewer FP or FN classifications, which would effect either precision or recall. However, as with the bias-variance trade-off 9, if the FP-rate is maximised this will affect the FN-rate, thus optimizing recall will have inverse proportional effect on precision, visa-versa. F1-Score 8 finds the harmonic mean between the two metrics, f1-scores are considered when the FP-rate and FN-rate are of equal weight in the performance measure of the model. The harmonic mean between precision and recall is preferred - as opposed to arithmetic or geometric - because it dissuades values that are disparate from each other and values that are undesirably low.

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{8}$$

eq. 8 : *represents the harmonic mean between precision* 7 *and recall* 6

F1-scores gives the average rate between recall and precision; better referred as 'harmonic mean'. Due to the specificity and accuracy often required in clinical domains, both recall and precision need to be considered, due to the high cost of misdiagnosis, or incorrect diagnosis incurs; leaving $F1$ as the final judiciary.

9.4 Cross-Validation

Cross validation(CV) is most commonly used as and evaluation method for SVMs 3.4. CV determines which partition of training, testing, and validation splits in the data used to train the model achieves the best performance. When considering CV, 3 types are typically considered: K-Fold Cross Validation, Stratified k-fold Cross Validation, and Leave One Out Cross Validation, where K-Folds is the most commonly used. (Shaikh (2018)). CV is used in traditional ML-methods and in NN architectures 5 to determine the optimal model performance based on the available data through iteration on different segments of the data, it also assist with tuning hyperparameters in NN to find the optimal performance, these hyperparamaters may include aspects of the NN architecture like the learning rate, epochs, number of layers and initialization weights - like identifying the word embeddings that achieves the best performance.

9.5 Entropy/Cross-Entropy

$$E_c[p, q] = \sum_{i=1}^{N} p(x_i) \log_2 q(x_i) \tag{9}$$

Entropy is often used as model evaluation methods in classification model and DL models. (Jincy-mol Joseph (2018)) Entropy has been used in thermodynamics extensively with adaptations in information sciences, especially when calculating the loss of information, further is has been used in data sciences to measure ML and DL models. Cross-Entropy is often used to determine accuracy in classification models, which is in itself a model performance measure(citation needed). If $p(x_i)$ from equation 9 is the

probability distribution of some observed event and $q(x_i)$ is the probability distribution determined by the given model on the same set of variables as $p(x_i)$, cross entropy 9 determines the average number of bits needed to accurately predict that event. Cross Entropy is determined by both Kullback-Leibler Divergence and Shannon Entropy, Kullback-Divergence measures the probability distributions between two(or more) events, where Shannon Entropy measures how much information are needed to effectively describe the probability of some event. Cross-entropy is mostly used as a *loss function* in classification models. What makes cross entropy different is that is suitable for back-propagation, which are essential for the memory function associated to RNNs 5.1, the final output is given as a probability between zero and one for a given class. This output format are what activation function achieves for RNN. Cross-entropy determines the variability between the distributions. The error - shown in eq. 9 - from cross-entropy is then summed and back-propagation is used to minimize the total error in the network. Cross-entropy is also preferred over other loss functions like Residual Sum of Squares (RSS) due the the heavy penalty in incurs for misclassifications.